

AD-A066 512

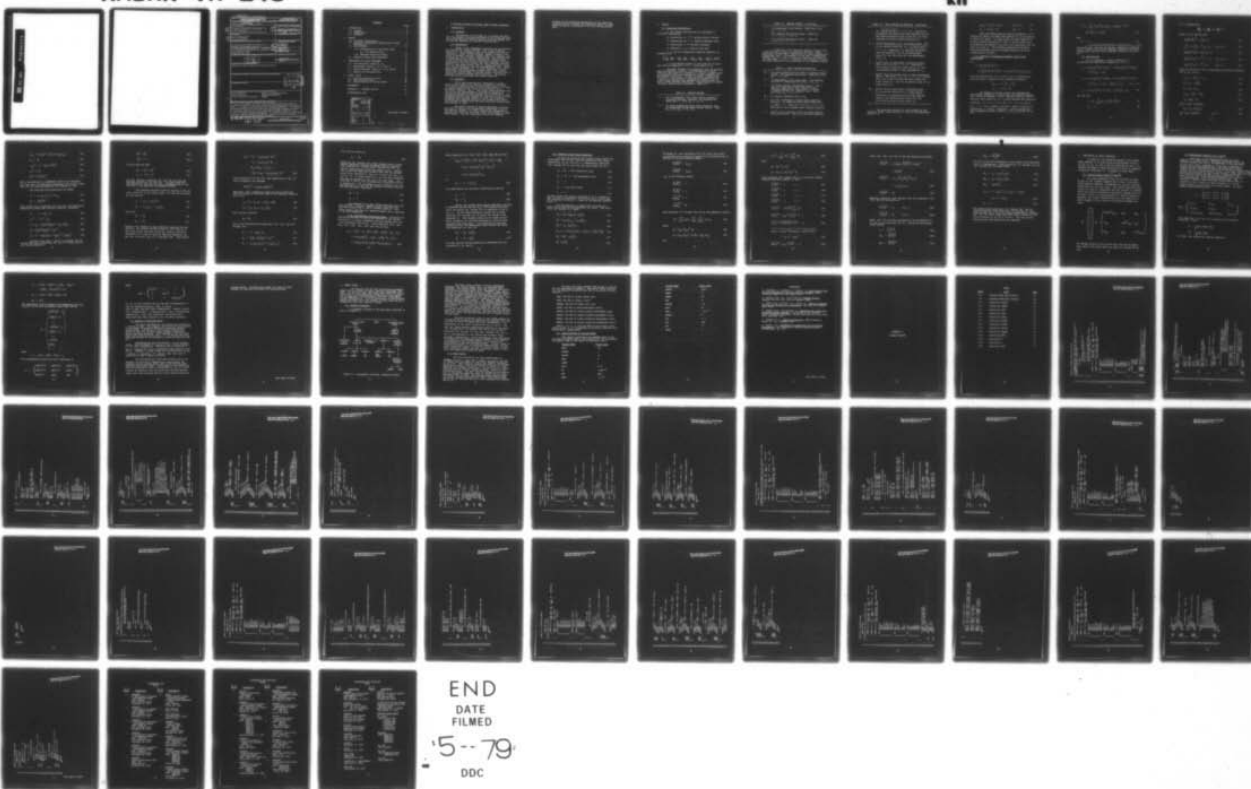
ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 17/9
A COMPUTER PROGRAM FOR DOUBLE SWEEP OPTIMAL SMOOTHING.(U)

UNCLASSIFIED

AMSAA-TR-246

| OF |

AD
A066 512



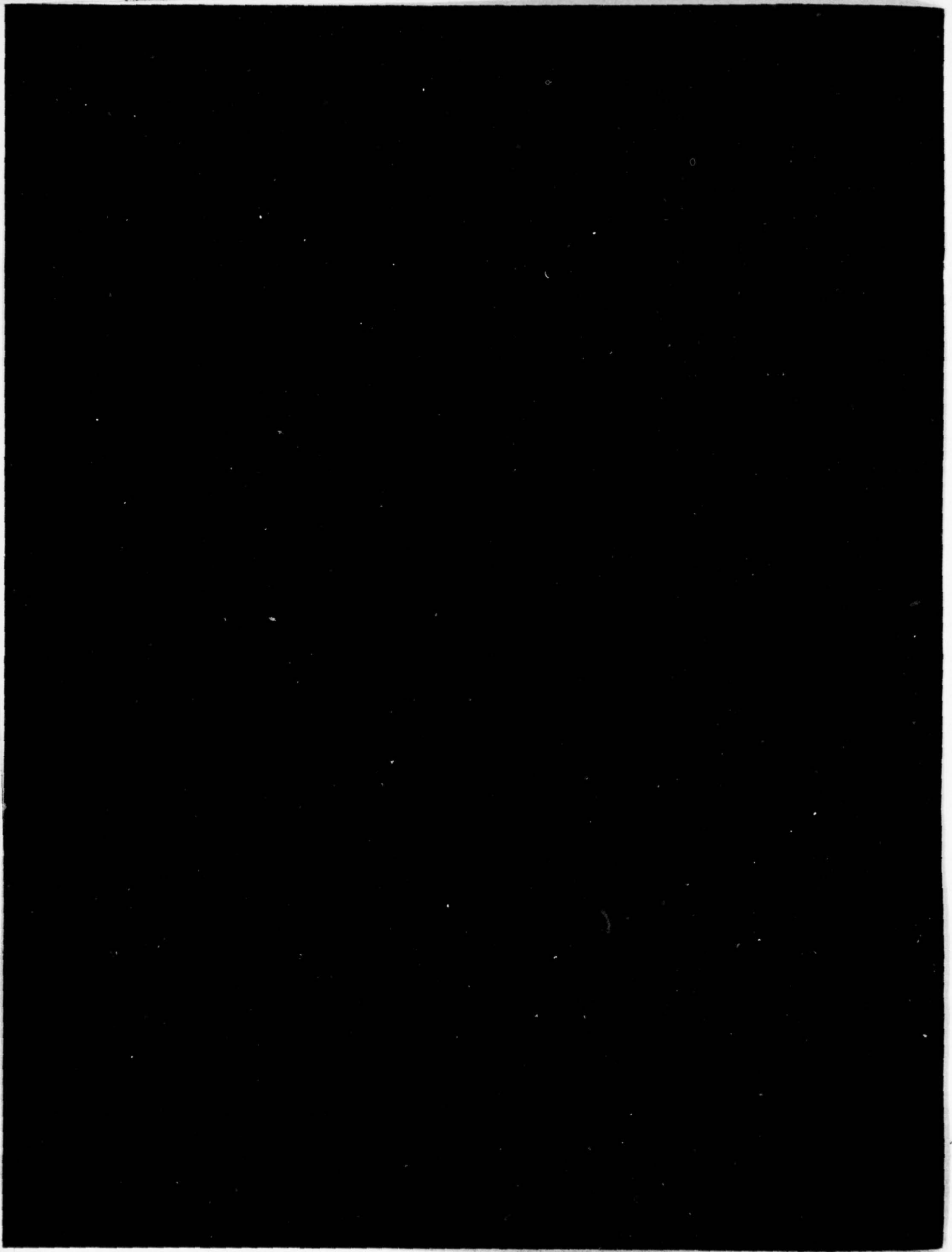
END
DATE
FILMED

5-79

DDC

DDC FILE COPY

AD A0 66512



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) AMSAA-TR- Technical Report No. 246	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Computer Program for Double Sweep Optimal Smoothing.	5. TYPE OF REPORT & PERIOD COVERED Technical / Rept.	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert A. Scheder	8. CONTRACT OR GRANT NUMBER(s)	
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Army Materiel Systems Analysis Activity Aberdeen Proving Ground, MD 21005	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA Project No. 1R765706M541	
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Materiel Development and Readiness Command 5001 Eisenhower Avenue Alexandria, VA 22333	12. REPORT DATE January 1979	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) (12) 74 p.	15. SECURITY CLASS. (of this report) UNCLASSIFIED	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Filter Evaluation Quadratic Programming Problem Optimal Estimation Double Sweep Smoothing Top Down Structured Programming FORTRAN		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report documents the double sweep smoothing program used at AMSAA to determine an aircraft's position, velocity, and acceleration from radar tracking data. It includes a tutorial guide to the underlying mathematics. A quadratic programming problem with linear constraints is formulated, reduced to a system of difference equations with initial and endpoint boundary conditions, and solved using Bryson's double sweep method. The iterative solution is programmed in simple, machine independent FORTRAN with top down structured programming. A listing is supplied and card decks are available.		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403 920

HB

CONTENTS

	Page
1. INTRODUCTION	5
1.1 Purpose	5
1.2 Background.	5
1.3 Approach.	5
2. THEORY	7
2.1 Principal Definitions	7
2.2 Quadratic Programming Problem (with Linear Constraints).	10
2.3 The Solution.	11
2.3a Determination of the Final State	
\hat{x}_n - The Kalman Filter.	13
2.3b Backward Correcting Sweep	18
2.4 Complete Double Sweep Equations	20
3. COMPUTATION OF INPUT VARIABLES	25
3.1 Initialization Inputs \bar{P}_0, \bar{X}_0	25
3.2 Measurement Inputs R, H, Z, and \bar{v}	26
3.3 Target Dynamical Inputs ϕ, Q, R , and \bar{W}	27
3.4 Remarks on Nonlinearities	30
4. USERS' GUIDE	33
4.1 Overall Program Structure	33
4.2 User Defined Inputs	34
4.3 Identification of Program Names	35
REFERENCES	37
APPENDIX A - PROGRAM LISTING	39
DISTRIBUTION LIST.	75

ACCESSION FOR	
NTIS	White Section <input checked="" type="checkbox"/>
DDC	Buff Section <input type="checkbox"/>
UNANNOUNCED	<input type="checkbox"/>
JUSTIFICATION.....	
BY.....	
DISTRIBUTION/AVAILABILITY CODES	
Dist.	AVAIL. and/or SPECIAL
A	

Next page is blank

79 03 27 002

A COMPUTER PROGRAM FOR DOUBLE SWEEP OPTIMAL SMOOTHING

1. INTRODUCTION

1.1 Purpose.

The purpose of this paper is to explain and document the computer routines developed by the author and used by the Army Materiel Systems Analysis Activity (AMSAA) for the optimal smoothing of aircraft tracking data.

1.2 Background.

AMSAA is the independent evaluator for the Division Air Defense Gun (DIVAD) program. One of the principal features of this program is the use of modern fire control techniques. These include optimal, Kalman filtering, routines to estimate aircraft (target) position, velocity, and acceleration from aircraft tracking data. One method used to study modern fire control consists of running target flight paths obtained during field tests through a digital simulation of the fire control and then comparing the estimates of the target state; that is, target position, velocity, and acceleration, with the corresponding "true" target state values (Reference 1). If this comparison is to be meaningful, the method used to obtain the "true" target state time history needs to provide greater accuracy than is possible to obtain through filtering alone. The method described below guarantees this needed accuracy.

1.3 Approach.

Field test tracking data consisting of target position as a function of time is processed in two steps. It is first filtered using a Kalman filter which is at least as sophisticated as the filter to be evaluated (e.g., nonlinear, adaptive, etc.). The target state estimates from the filter are then optimally corrected by sweeping in reverse chronological order through the tracking data. The corrected estimates are necessarily superior to the filtered estimates since at each instant of time, the corrected estimates were computed using all the tracking data while the filtered estimates were derived using only tracking data from earlier in time.

The general theory of optimal smoothing as considered here is fairly well established (References 2 and 3). The details, however, are often outside the background of analysts who could use (and have used) this program in other Army studies. Section 2 provides a brief but complete

tutorial of the underlying mathematics of the smoothing program which will enable the interested reader to understand and modify the program to fill his own particular need.

The purpose of this paper is to explain and demonstrate the computer simulation of aircraft tracking data developed by the Army Research Office-Durham (AROD) for the computer simulation of aircraft tracking data.

2. BACKGROUND

AROD is the independent evaluation for the Division of Defense and (DDIVAD) program. One of the primary features of this program is the use of modern line control techniques. These include: (1) Kalman filtering, (2) time to estimate aircraft position, velocity, and acceleration from aircraft tracking data. The method used to track modern line control consists of tracking error signals between the line control and the aircraft. The state of the line control and the aircraft position, velocity, and acceleration, with the corresponding "error" signal, are used to obtain the "error" signal. If this comparison is to be meaningful, the method used to obtain the "error" signal must take history into account. The method used to obtain the "error" signal is the method described below. This method is necessary.

2.1. Approach

Figure 1 shows the data collection of target position as a function of time. The data is presented in two steps. It is first filtered using a Kalman filter which is an linear as sophisticated as the filter can be estimated. The nonlinear, discrete, and the linear state estimates from the filter are then optimally combined by weighted in reverse chronological order to obtain the tracking data. The corrected estimates are necessary to correct the filter. The corrected estimates are at each instant of time. The corrected estimates were computed using the tracking data while the corrected estimates were computed using only tracking data from earlier to time.

The general theory of optimal smoothing is described in this paper. It is briefly outlined in Appendix 1 and 2. The details, however, are given in the Appendix of the program which will be used. This program is then analyzed and modified to fill his own particular need. Section 1 provides a brief but complete

2. THEORY

2.1 Principal Definitions.

- All vectors and matrices are expressed in cartesian coordinates.
- Superscript -1, (⁻¹), denotes matrix inverse.
- Superscript bar, ([¯]), denotes expected value.
- Superscript T, (^T) denotes transpose.
- Subscript i, (_i), denotes time step.
- ∇ , the 3n+1 dimensional gradient operator, is defined to be

$$\left(\frac{\partial}{\partial x_0}, \frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n}, \frac{\partial}{\partial w_0}, \dots, \frac{\partial}{\partial w_{n-1}}, \frac{\partial}{\partial v_1}, \dots, \frac{\partial}{\partial v_n} \right); \text{ where}$$

n is the maximum number of time steps or the maximum value of subscript i (i.e., i goes from 0 to n).

The basic problem is to determine from the tracking data the position, velocity, and acceleration or state vector of the target as a function of time in a manner which is consistent with our knowledge of aircraft dynamics and measurement uncertainty. As will be shown in Subsections 2.2 and 2.3, the process of determining estimates of the target state requires that estimates of several other vectors be simultaneously determined. These vectors are termed "derived" vectors and are defined in Table 2-1.

TABLE 2-1 DERIVED VECTORS

-
- X : The 9 dimensional (9-D) state vector consisting of target position, velocity, and acceleration. There are n + 1 x's, x_i , $i = 0 \dots n$.
- W : 9-D state transition error vector which can also be interpreted as the state control vector. There are n W's, w_i , $i = 0 \dots n-1$.
-

TABLE 2-1 DERIVED VECTORS (Continued)

-
- V : 3-D measurement error vector. There are n V 's, V_i , $i = 1...n$.
- λ : 9-D, Lagrange Multiplier vector. There are $n + 1$ λ 's, λ_i , $i = 0...n$.
- μ : 3-D, Lagrange Multiplier vector. There are n μ 's, μ_i , $i = 1...n$.
-

In addition to the "derived" vectors, we will need to define several "input" vectors and matrices. These quantities are mathematic expressions of target dynamics and measurement error. The analysis in the next 3 subsections assumes that the "input" vectors and matrices are known for all time steps, $i = 0...n$. Section 3 will explain in detail how these quantities are constructed in the computer program.

TABLE 2-2 INPUT VECTORS AND MATRICES

-
- \bar{W} : 9-D state transition error mean (or nominal control) vector. The expected value of W . There are n \bar{W} 's, \bar{W}_i , $i = 0...n-1$.
- \bar{V} : 3-D measurement error mean vector. The expected value of V . There are n \bar{V} 's, \bar{V}_i , $i = 1...n$.
- Z : 3-D target position measurement vector. The Z data as a function of time, obtained from a tracking radar during field tests, is stored in digital format on a computer tape. There are n Z 's, Z_i , $i = 1...n$.
- \bar{X}_0 : 9-D apriori expected value of X_0 .
- Φ : $9 \times 9-D^2$, nonsingular, target state transition matrices. Φ_i takes the state vector from time i to time $i + 1$. There are n Φ 's, Φ_i , $i = 0...n-1$. Physically, the Φ_i embody basic aircraft dynamics.
- Q : $9 \times 9-D^2$ state transition error (or state control) covariance matrix. The Q_i are positive definite;
-

TABLE 2-2 INPUT VECTORS AND MATRICES (Continued)

- $Q_i = \overline{(W_i - \bar{W}_i)(W_i - \bar{W}_i)^T}$, $i = 0 \dots n-1$. Physically, Q_i measures the amount that the expected value of the state transition error vector, \bar{W}_i , can vary from the true state transition error, W_i .
- R : $3 \times 3 - D^2$ measurement error covariance matrix. The R_i are positive definite; $R_i = \overline{(V_i - \bar{V}_i)(V_i - \bar{V}_i)^T}$, $i = 1 \dots n$. Physically, R_i measures the amount that the expected value of the measurement error, \bar{V}_i , can vary from the true measurement error, V_i .
- H : $3 \times 9 - D^2$ state to measurement transition matrix. There are n H's, H_i , $i = 1 \dots n$. Physically, the H_i matrix takes the state vector, X_i , into the coordinate frame of the measurements, Z_i .
- Γ : $9 \times 9 - D^2$ state transition error to state transition matrix, There are n Γ 's, Γ_i , $i = 0 \dots n-1$. Physically, the Γ_i matrix takes the state transition error vector, W_i , into the coordinate frame of the state vector X_i .
- \bar{P}_0 : $9 \times 9 - D^2$ initial state error covariance matrix. \bar{P}_0 is positive definite; $\bar{P}_0 = \overline{(X - \bar{X}_0)(X - \bar{X}_0)^T}$. Physically, \bar{P}_0 measures how much the expected value of the initial state vector, \bar{X}_0 , can vary from the true initial state, X_0 .

The above definitions are tied together by the dynamical target state equation (1) and by the measurement equation (2).

$$X_{i+1} = \Phi_i X_i + \Gamma_i W_i \quad i=0 \dots n-1 \quad (1)$$

$$Z_i = H_i X_i + V_i \quad i=1 \dots n \quad (2)$$

From the definitions of Q and R we expect W_i to be close to \bar{W}_i when Q_i is small and similarly V_i to be close to \bar{V}_i when R_i is small. In other words, we seek estimates of X_i which satisfy equations (1) and (2) and for which the corresponding estimates of W_i and V_i are consistent with the size of Q_i and R_i . The following formulation satisfies these requirements and has the added advantage of being mathematically solvable.

2.2 Quadratic Programming Problems (with Linear Constraints).

Let

$$L = 1/2 (X_0 - \bar{X}_0)^T P_0^{-1} (X_0 - \bar{X}_0) + \\ 1/2 \sum_{i=0}^{n-1} [(W_i - \bar{W}_i)^T Q_i^{-1} (W_i - \bar{W}_i) + (V_{i+1} - \bar{V}_{i+1})^T R_{i+1}^{-1} (V_{i+1} - \bar{V}_{i+1})]$$

find the estimates $\hat{X}_i, \hat{W}_i, \hat{V}_i$ of X_i, W_i, V_i respectively which minimize L subject to the constraint equations

$$\begin{aligned} \hat{X}_{i+1} &= \Phi_i \hat{X}_i + \Gamma_i \hat{W}_i \\ Z_i &= H_i \hat{X}_i + \hat{V}_i \end{aligned} \quad (3)$$

The Theory of Optimal Control and Mathematical Programming (Reference 3) proves that this type of problem has a solution whenever it has a feasible solution, that is, whenever there exists $\hat{X}_i, \hat{W}_i, \hat{V}_i$ which satisfy the constraint equations. $\hat{X}_i = 0, i = 0 \dots n, \hat{W}_i = 0, i = 0 \dots n,$ and $\hat{V}_i = Z_i, i = 1 \dots n$ is a feasible solution. Hence a solution to formulation (3) exists. Reference 3 further proves that there necessarily exists vectors λ_i and μ_i such that if we let

$$J = L + \sum_{i=1}^n [\lambda_{i-1}^T (\phi_{i-1} X_{i-1} + \Gamma_{i-1} W_{i-1} - X_i) + \mu_i^T (H_i X_i + V_i - Z_i)] \quad (4)$$

Then

$$\nabla J = 0. \quad (5)$$

J is called the Hamiltonian function and λ_i, μ_i are the traditional Lagrange multipliers (Reference 2 supplies proof). Since L is a convex function, equation (5) is also a sufficient condition for a feasible solution to be optimal.

2.3 The Solution.

We can highlight a useful property of J by defining the Hamiltonian sequence $J_i, i=0\dots n$.

$$\begin{aligned} J_0 &= 1/2 (X_0 - \bar{X}_0)^T \bar{P}_0^{-1} (X_0 - \bar{X}_0) + 1/2 (W_0 - \bar{W}_0)^T \bar{Q}_0^{-1} (W_0 - \bar{W}_0) \\ &\quad + \lambda_0^T (\phi_0 X_0 + \Gamma_0 W_0) \\ J_i &= 1/2 (W_i - \bar{W}_i)^T \bar{Q}_i^{-1} (W_i - \bar{W}_i) + 1/2 (V_i - \bar{V}_i)^T \bar{R}_i^{-1} (V_i - \bar{V}_i) \\ &\quad + \lambda_i^T (\phi_i X_i + \Gamma_i W_i) + \mu_i^T (H_i X_i + V_i) \quad (i=1\dots n-1) \\ J_n &= 1/2 (V_n - \bar{V}_n)^T \bar{R}_n^{-1} (V_n - \bar{V}_n) + \mu_n^T (H_n X_n + V_n) \end{aligned} \quad (6)$$

We find that

$$J = J_0 + \sum_{i=1}^n [J_i - \lambda_{i-1}^T X_i - \mu_i^T Z_i] \quad (7)$$

$$\lambda_n = 0. \quad (8)$$

$\nabla J = 0$ implies that

$$\frac{\partial J}{\partial X_i} = 0, \quad \frac{\partial J}{\partial W_i} = 0, \quad \frac{\partial J}{\partial V_i} = 0$$

which in turn implies that

$$(\hat{X}_0 - \bar{X}_0)^T \bar{P}_0^{-1} + \lambda_0^T \Phi_0 = 0 \quad (9)$$

$$\lambda_i^T \Phi_i + \mu_i^T H_i - \lambda_{i-1}^T = 0 \quad i=1 \dots n-1 \quad (10)$$

$$(\hat{W}_0 - \bar{W}_0)^T Q_0^{-1} + \lambda_0^T \Gamma_0 = 0 \quad (11)$$

$$(\hat{W}_i - \bar{W}_i)^T Q_i^{-1} + \lambda_i^T \Gamma_i = 0 \quad i=1 \dots n-1 \quad (12)$$

$$(\hat{V}_i - \bar{V}_i)^T R_i^{-1} + \mu_i^T = 0 \quad i=1 \dots n \quad (13)$$

Hence we arrive at a set of simultaneous difference equations for \hat{X}_i , \hat{W}_i , and \hat{V}_i .

$$\lambda_{i-1} = H_i^T \mu_i + \Phi_i^T \lambda_i \quad (14)$$

$$\hat{W}_i = \bar{W}_i - Q_i \Gamma_i^T \lambda_i \quad (15)$$

$$\hat{V}_i = \bar{V}_i - R_i \mu_i \quad (16)$$

$$\hat{X}_{i+1} = \Phi_i \hat{X}_i + \Gamma_i \hat{W}_i \quad (17)$$

$$Z_i = H_i \hat{X}_i + \hat{V}_i \quad (18)$$

with initial condition;

$$\hat{X}_0 = \bar{X}_0 - \bar{P}_0 \Phi_0^T \lambda_0 \quad (19)$$

and final condition

$$\lambda_n = 0 \quad (20)$$

These equations combine to yield our fundamental system of equations;

$$\hat{X}_{i+1} = \Phi_i \hat{X}_i - \Gamma_i Q_i \Gamma_i^T \lambda_i + \Gamma_i \bar{W}_i \quad (21)$$

$$\lambda_{i-1} = \Phi_i^T \lambda_i + H_i^T R_i^{-1} H_i \hat{X}_i - H_i^T R_i^{-1} (Z_i - \bar{V}_i) \quad (22)$$

with boundary conditions

$$\hat{X}_0 = \bar{X}_0 - \bar{P}_0 \Phi_0^T \lambda_0 \quad (23)$$

$$\lambda_n = 0 \quad (24)$$

Hence the quadratic programming problem, (3), is reduced to solving equations (21) and (22) with boundary conditions (23) and (24). Note all other derived vectors in (14) through (20) can be expressed in terms of \hat{X}_i , λ_i , and input variables. For example, $\mu_i = -R_i^{-1} (Z_i - H_i \hat{X}_i - \bar{V}_i)$.

The difficulty in solving equations (21) and (22) is that the boundary conditions, (23) and (24), are split -- one initial condition and one end point or final condition. Reference 4 suggests a technique for obtaining the final value, \hat{X}_n , of the optimal solution to equations (21) through (24). Having determined \hat{X}_n , we replace the split conditions (23) and (24) with the two end point conditions and can then find \hat{X}_i for all i by iterating equations (21) and (22) backwards in time. The mathematics are carried out in the next two subsections.

2.3a Determination of the Final State \hat{X}_n - The Kalman Filter. We first establish certain properties of a homogenous solution X_i^h , and a non-homogenous particular solution, X_i^p , whose sum is the optimal solution \hat{X}_i (i.e., $X_i^h + X_i^p = \hat{X}_i$ for $i = 0 \dots n$). We do not actually compute X_i^h or X_i^p but rather use certain of their analytical

properties to derive recursive relations which will ultimately yield the end point vector, \hat{X}_n .

For notational convenience we introduce the additional variables A_i and B_i .

$$A_i = H_i^T R_i^{-1} H_i \quad (25)$$

$$B_i = \Gamma_i Q_i \Gamma_i^T \quad (26)$$

Note the A_i and B_i are composed of input matrices and are therefore known for all i .

Consider the homogenous problem;

$$X_{i+1}^h = \Phi_i X_i^h - B_i \lambda_i^h \quad (27)$$

$$\lambda_{i-1}^h = \Phi_i^T \lambda_i^h + A_i X_i^h \quad (28)$$

with boundary conditions

$$X_0^h = -\bar{P}_0 \Phi_0^T \lambda_0^h \quad (29)$$

$$\lambda_n^h = -\lambda_n^p. \quad (30)$$

The form of equations (25) through (29) suggests the relation

$$X_i^h = -P_i \Phi_i^T \lambda_i^h \quad i=0\dots n \quad (31)$$

This is certainly true for $i = 0$. The P_i for $i > 0$ are proportionately matrices still to be determined. Substituting relation (31) into equations (27) and (28), we arrive directly at uncoupled equations for P_i , λ_i^h , and X_i^h .

$$P_{i+1} = [\phi_i P_i \phi_i^T + B_i] [I - A_{i+1} P_{i+1}] \quad (32)$$

$$P_0 = \bar{P}_0 \quad (33)$$

$$\lambda_{i-1}^h = [I - A_i P_i] \phi_i^T \lambda_i^h \quad (34)$$

$$\lambda_n^h = -\lambda_n^p \quad (35)$$

$$X_i^h = -P_i \phi_i^T \lambda_i^h \quad (36)$$

The P_i matrices play a fundamental role in our analysis.

Note that they are completely determined from initial condition (33) by iterating equation (32) forward in time.

The following definitions will be useful:

$$\tilde{P}_i = \phi_{i-1} P_{i-1} \phi_{i-1}^T + B_{i-1} \quad (37)$$

$$K_i = P_i H_i^T R_i^{-1} \quad (38)$$

These definitions, equations (32) and (33), and some matrix manipulation yields the following important relations;

$$P_i = [I - P_i A_i] \tilde{P}_i \quad (39)$$

$$P_i^{-1} = \tilde{P}_i^{-1} + A_i \quad (40)$$

$$P_i = \tilde{P}_i - \tilde{P}_i H_i^T [H_i \tilde{P}_i A_i^T + R_i]^{-1} H_i \tilde{P}_i \quad (41)$$

$$K_i = \tilde{P}_i H_i^T [H_i \tilde{P}_i H_i^T + R_i]^{-1} \quad (42)$$

$$P_i = (I - K_i H_i) \tilde{P}_i (I - K_i H_i)^T + K_i R_i K_i^T \quad (43)$$

We shall also find it useful to consider the particular solution to equations (21) and (22) which satisfy the initial condition

$$x_0^p = \bar{x}_0 \quad (44)$$

$$\lambda_0^p = 0 \quad (45)$$

In this case the sums

$$\hat{x}_i = x_i^h + x_i^p \quad (46)$$

$$\lambda_i = \lambda_i^h + \lambda_i^p \quad (47)$$

not only satisfy equations (21) and (22) but also the boundary conditions (23) and (24). In other words the sums defined in (46) and (47) are a representation of the solution to the quadratic programming problem.

The promised recursion equation leading to the end point vector \hat{x}_n is now within reach. To this end we define the new vectors;

$$y_i = x_i^p + p_i \phi_i^T \lambda_i^p \quad (48)$$

$$\tilde{y}_i = \phi_{i-1} y_{i-1} + \Gamma_{i-1} \bar{w}_{i-1} \quad (49)$$

Note that

$$y_0 = \bar{x}_0 \quad (50)$$

$$y_n = \hat{x}_n \quad (51)$$

Equation (51) assures us that finding an equation for computing y_i (by forward iteration from y_0) will lead us to the value of the end point vector \hat{x}_n . Such an iterative expression for y_i can be found by applying equations (21) and (22) to the x_i^p and λ_i^p in equation (48). This yields

$$\begin{aligned}
Y_{i+1} = & [\phi_i - P_{i+1} A_{i+1} \phi_i] X_i^P + \\
& [\Gamma_i - P_{i+1} A_{i+1} \Gamma_i] \bar{W}_i + \\
& K_{i+1} [Z_{i+1} - \bar{V}_{i+1}] + \\
& [-B_i + P_{i+1} + P_{i+1} A_{i+1} B_i] \lambda_i^P \quad (52)
\end{aligned}$$

Using relations (37) and (39), the coefficient of the λ_i^P term in equation (52) becomes

$$[\phi_i P_i \phi_i^T - P_{i+1} A_{i+1} \phi_i P_i \phi_i^T]$$

Therefore, after recombining terms and using definition (49), we arrive at the final form of the iterative equation for the Y_i ;

$$Y_i = \tilde{Y}_i + K_i [Z_i - H_i \tilde{Y}_i] - K_i \bar{V}_i \quad (53)$$

$$\tilde{Y}_i = \phi_{i-1} Y_{i-1} + \Gamma_{i-1} \bar{W}_{i-1} \quad (54)$$

with initial condition

$$Y_0 = \bar{X}_0 \quad (55)$$

the K_i are determined from equations (32), (33), and (37) through (43);

$$P_i = (I - K_i H_i) \tilde{P}_i \quad (56)$$

$$K_i = \tilde{P}_i H_i^T [H_i \tilde{P}_i H_i^T + R_i]^{-1} \quad (57)$$

$$\tilde{P}_i = \phi_{i-1} P_{i-1} \phi_{i-1}^T + \Gamma_{i-1} Q_i \Gamma_{i-1}^T \quad (58)$$

with initial condition

$$P_0 = \bar{P}_0 \quad (59)$$

Equations (53) through (59) taken together form a closed system completely solvable by simple forward iteration. They comprise our "forward sweep" through the data, Z_i , and are traditionally called the Kalman filter. Clearly, the Y_k have the important property that they are optimal target state estimates at time $i=k$ if given only data, Z_i , for $i=0\dots k$. In other words the Y_i are the best estimates that can be obtained on line (i.e., in real time). The variables Y_i and P_i will be physically interpreted further in section 2.4. For the moment we are concerned with solving equations (21) and (22) by backward iteration starting with the end point conditions

$$\hat{X}_n = Y_n \quad (60)$$

$$\lambda_n = 0 \quad (61)$$

One approach is to just iterate equations (21) and (22) directly. Another, computationally more efficient approach utilizes the Y_i and P_i already computed. This latter method comprises the "Backward Sweep" and is detailed in the next subsection.

2.3b The Backward Correcting Sweep. Utilizing the preceeding analysis we can arrive directly at the optimal solution \hat{X}_i , λ_i using a backward iteration scheme where the λ_i are uncoupled from the \hat{X}_i . Using equations (22), (39), (40), (41), (42), (43), (48), and (53) we find

$$\begin{aligned} \lambda_{i-1} &= \phi_i^T \lambda_i + A_i [X_i^h + X_i^p] - H_i^T R_i^{-1} [Z_i - \bar{V}_i] \\ &= [I - A_i P_i] \phi_i^T \lambda_i + A_i Y_i - H_i^T R_i^{-1} (Z_i - \bar{V}_i) \\ &= [I - A_i P_i] [\phi_i^T \lambda_i - H_i^T R_i^{-1} (Z_i - \bar{V}_i - H_i Y_i)]. \end{aligned} \quad (62)$$

Using equations (21), (34), (37), (39), and (48) we find

$$\begin{aligned}\hat{X}_{i+1} &= \Phi_i(X_i^h + X_i^p) - B_i(\lambda_i^h + \lambda_i^p) + \Gamma_i \bar{W}_i \\ &= Y_{i+1} - P_{i+1} \Phi_{i+1}^T \lambda_{i+1}^p - \tilde{P}_{i+1} \lambda_i^h \\ &= Y_{i+1} - P_{i+1} \Phi_{i+1}^T \lambda_{i+1}\end{aligned}$$

or

$$\hat{X}_i = Y_i - P_i \Phi_i^T \lambda_i \quad (63)$$

For completeness the end point conditions are written

$$\lambda_n = 0 \quad (64)$$

$$\hat{X}_n = Y_n \quad (65)$$

Hence, the "Double Sweep Optimal Smoother" consists of sweeping through the data Z_i in the forward direction using the system (53) through (59) to determine P_i and Y_i $i=0 \dots n$. We then sweep backwards through the Z_i using equations (62) through (65) to obtain the optimal state estimates \hat{X}_i . This backward sweep can be viewed as optimally correcting the filtered estimates Y_i with the data received after time step i . The estimates of the state transition error and measurement error vectors are found from equations (15) and (16)

$$\hat{W}_i = \bar{W}_i - Q_i \Gamma_i^T \lambda_i \quad (66)$$

$$\hat{V}_i = Z_i - H_i \hat{X}_i. \quad (67)$$

The next section derives measures of confidence for the estimates \hat{X}_i , \hat{W}_i , and \hat{V}_i .

2.4 Complete Double Sweep Equations.

If the true values of the initial state vector, the measurement error vector, and the transition error vector equalled \bar{X}_0 , \bar{V}_i , \bar{W}_i (for all i), respectively, equations (14) through (20) or (21) and (22) would imply for all i ,

$$\hat{W}_i = \bar{W}_i = \text{true transition error} \quad (68)$$

$$\hat{V}_i = \bar{V}_i = \text{true measurement error} \quad (69)$$

$$\lambda_i = 0 \quad (70)$$

$$\hat{X}_i = \text{true state vector} \quad (71)$$

$$L = 0 \quad (72)$$

In other words, the optimal estimates \hat{X} , \hat{W} , \hat{V} , necessarily take on their corresponding true values since in this case the performance criteria, L , takes on its smallest possible value, 0.

Let the operator, δ , denote the variation of a vector value from the corresponding true value. Then equations (62) through (72) imply,

$$\delta Y_i = \delta \tilde{Y}_i - K_i H_i \delta \tilde{Y}_i - K_i \delta \bar{V}_i \quad (73)$$

$$\delta \tilde{Y}_i = \Phi_{i-1} \delta Y_{i-1} + \Gamma_{i-1} \delta \bar{W}_{i-1} \quad (74)$$

$$\delta \hat{X}_i = \delta Y_i - P_i \Phi_i^T \delta \lambda_i \quad (75)$$

$$\delta \lambda_{i-1} = [I - A_i P_i] [\Phi_i \delta \lambda_i + A_i \delta Y_i + H_i^T R_i^{-1} \delta \bar{V}_i] \quad (76)$$

$$\delta \hat{W}_i = \delta \bar{W}_i - Q_i \Gamma_i^T \delta \lambda_i \quad (77)$$

$$\delta \hat{V}_i = H_i \delta \hat{X}_i \quad (78)$$

We assume that the variations of \bar{V}_i , \bar{W}_i and \bar{X}_0 from their corresponding true values are given as in the definitions of Section 2.1 and are entirely random;

$$\overline{\delta \bar{V}_i \delta \bar{V}_i^T} = R_i \delta_{ij} \quad (79)$$

$$\overline{\delta \bar{W}_i \delta \bar{W}_j^T} = Q_i \delta_{ij} \quad (80)$$

(δ_{ij} is the Kronecker delta)

$$\overline{\delta \bar{V}_i \delta \bar{W}_j^T} = 0 \quad (81)$$

$$\overline{\delta \bar{V}_i \delta \bar{X}_0^T} = 0 \quad (82)$$

$$\overline{\delta \bar{W}_i \delta \bar{X}_0^T} = 0 \quad (83)$$

$$\overline{\delta \bar{X}_0 \delta \bar{X}_0^T} = \bar{P}_0 \quad (84)$$

From equations (73) through (84) we get the important results,

$$\delta Y_i = \prod_{k=0}^{i-1} C_k \delta Y_0 + \sum_{k=0}^{i-1} \left(\prod_{\ell=k+1}^{i-1} C_\ell \right) c_k \quad (85)$$

where

$$C_k = P_{k+1} \tilde{P}_{k+1}^{-1} \phi_k \quad (86)$$

$$c_k = P_{k+1} \tilde{P}_{k+1}^{-1} \Gamma_k \bar{W}_k - K_{k+1} \bar{V}_{k+1} \quad (87)$$

and

$$\delta \lambda_i^T = \sum_{\ell=i+1}^n d_{\ell}^T \left(\prod_{k=i+1}^{\ell-1} D_k \right) \quad (88)$$

where

$$D_k = \tilde{P}_k^{-1} P_k \Phi_k^T \quad (89)$$

$$d_k = A_k Y_k + H_k^T R_k^{-1} \bar{V}_k \quad (90)$$

Using equations (85) through (90) it is straight forward, although sometimes lengthy, to show;

$$\overline{\delta Y_i \delta \bar{V}_{\ell}^T} = 0 \quad \ell > i \quad (91)$$

$$\overline{\delta \tilde{Y}_i \delta \bar{V}_{\ell}^T} = 0 \quad \ell \geq i \quad (92)$$

$$\overline{\delta Y_i \delta \bar{W}_{\ell}^T} = 0 \quad \ell \geq i \quad (93)$$

$$\overline{\delta \tilde{Y}_i \delta \bar{W}_{\ell}^T} = 0 \quad \ell \geq i \quad (94)$$

$$\overline{\delta Y_i \delta Y_{\ell}^T} = \overline{\delta Y_i \delta Y_i^T} \prod_{k=1}^{\ell-1} C_k^T \quad (95)$$

$$\overline{\delta Y_i \delta \lambda_i^T} - P_i \Phi_i^T \overline{\delta \lambda_i \delta \lambda_i^T} = 0 \quad (96)$$

$$\overline{(A_i \delta \lambda_i + H_i^T R_i^{-1} \delta \bar{V}_i) \delta \lambda_i^T} = 0 \quad (97)$$

$$\begin{aligned} & \overline{[A_i \delta \lambda_i + H_i^T R_i^{-1} \delta \bar{V}_i] [A_i \delta Y_i + H_i^T R_i^{-1} \delta \bar{V}_i]^T} \\ & = A_i (I - P_i A_i) \end{aligned} \quad (98)$$

$$\overline{\delta \bar{W}_{\ell} \delta \lambda_i^T} - Q_i \Gamma_i^T \overline{\delta \lambda \delta \lambda^T} = 0 \quad (99)$$

Using (91), (92), and (95) we get the recursive relations;

$$\begin{aligned} \overline{\delta Y_i \delta Y_i^T} &= (I - K_i H_i) \overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} (I - K_i H_i)^T \\ &\quad + K_i R_i K_i^T \end{aligned} \quad (100)$$

$$\begin{aligned} \overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} &= \Phi_{i-1} \overline{\delta Y_{i-1} \delta Y_{i-1}^T} \Phi_{i-1}^T \\ &\quad + \Gamma_{i-1} Q_{i-1} \Gamma_{i-1}^T \end{aligned} \quad (101)$$

$$\overline{\delta Y_0 \delta Y_0^T} = \bar{P}_0 \quad (102)$$

Comparing equations (100) through (102) with equations (33), (37), and (43) we arrive at

$$\overline{\delta Y_i \delta Y_i^T} = P_i \quad i=0 \dots n \quad (103)$$

$$\overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} = \tilde{P}_i \quad i=1 \dots n \quad (104)$$

Hence, the P (\tilde{P}) are the covariance of the variations in Y (\tilde{Y}) due to the variations $\delta \bar{V}$, $\delta \bar{W}$. Continuing along these lines, define;

$$\Lambda_i = \overline{\delta \lambda_i \delta \lambda_i^T} \quad (105)$$

$$SP_i = \overline{\delta \hat{X}_i \delta \hat{X}_i^T} \quad (106)$$

$$SQ_i = \overline{\delta \hat{W}_i \delta \hat{W}_i^T} \quad (107)$$

$$SR_i = \overline{\delta \hat{V}_i \delta \hat{V}_i^T} \quad (108)$$

SP, SQ, SR are the covariances of the errors in the estimates \hat{X} , \hat{W} , \hat{V} , respectively. Using equations (73) through (78) and (91) through (99) we easily find;

$$SP_i = P_i - P_i \phi_i^T \Lambda_i \phi_i P_i \quad (109)$$

$$SQ_i = Q_i - Q_i \Gamma_i \Lambda_i \Gamma_i^T Q_i \quad (110)$$

$$SR_i = H_i SP_i H_i^T \quad (111)$$

$$\Lambda_{i-1} = (I - P_i A_i)^T \phi_i^T \Lambda_i \phi_i (I - P_i A_i) + A_i (I - P_i A_i) \quad (112)$$

$$\Lambda_n = 0 \quad (113)$$

The forward sweep equations, (53) through (59), and the complete backward sweep equations, (62) through (67) and (109) through (113), form the "Double Sweep Optimal Smoother." Note that the double sweep equations require values for the input variables \bar{X}_0 , \bar{P}_0 , \bar{W} , \bar{V} , R and Q. The next section describes how these quantities are computed.

3. COMPUTATION OF INPUT VARIABLES

In order for the iterative process of the double sweep smoother to be initialized and maintained, the \bar{X}_0 , \bar{P}_0 , \bar{W}_i , \bar{V}_i , R_i , Q_i , and ϕ_i must be supplied for all i . A complete analysis of these quantities is given in "Adaptive Estimation" (Reference 5). A summary is given below.

3.1 Initialization Inputs \bar{P}_0 and \bar{X}_0 .

All nine components of \bar{X}_0 are set equal to zero. \bar{P}_0 has all off diagonal elements set to zero. Its first three diagonal elements, corresponding to initial position uncertainty are set very large. The initial velocity uncertainty is large but recognizes that the aircraft is subsonic while the initial acceleration uncertainty assumes that the aircraft is not in a severe maneuver at initial detection. Smoother (and filter) estimates are not very sensitive to small changes in \bar{P}_0 as long as the diagonal elements are large.

$$\bar{X}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \bar{P}_0 = \begin{pmatrix} (3000)^2 I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & (240)^2 I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & (25)^2 I_{3 \times 3} \end{pmatrix}$$

The assumed values of the initial state and the corresponding blocks of \bar{P}_0 given above can easily be changed by the user.

3.2 Measurement Inputs R, H, Z, and \bar{V} .

Since most of the tracking data that we use is supplied in a fixed inertial cartesian coordinate frame, the smoother assumes that the Z_i are expressed in the same frame as the state vector, although it can be easily modified to account for any measurement frame. The model of the measurement errors in the program accounts for constant variance angular errors and also constant variance linear, glint type, errors in both azimuth and elevation and for a fixed variance linear error in range. The covariance matrix of these errors is computed in the radar line of sight frame and then expressed, via a similarity transformation, in the fixed reference frame. The reference to sensor frame coordinate transformation TRS_i is computed at each i using the latest estimate, Y_i , of the target position vector during the forward sweep. The measurement related matrices are;

$$H_i = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$TRS_i = \begin{pmatrix} 1_{Y \div |Y|} & 2_{Y \div |Y|} & 3_{Y \div |Y|} \\ -2_{Y \div |Yg|} & 1_{Y \div |Yg|} & 0 \\ -1_Y 3_{Y \div |Y| |Yg|} & -2_Y 3_{Y \div |Y| |Yg|} & |Yg| \div |Y| \end{pmatrix}$$

(the subscript i 's on the Y 's are omitted for notational convenience). Where

$$|Y| = \sqrt{1_Y^2 + 2_Y^2 + 3_Y^2}$$

$$|Yg| = \sqrt{1_Y^2 + 2_Y^2}$$

and upper left superscript denotes component.

$$R_i = TRS_i^T \begin{pmatrix} \text{sigrxs}_i & 0 & 0 \\ 0 & \text{sigrys}_i & 0 \\ 0 & 0 & \text{sigrzs}_i \end{pmatrix} TRS_i$$

where

$$\text{sigrxs}_i = \text{sigmr}^2$$

$$\text{sigrys}_i = (\text{stdaz1})^2 + (|Y_i| \text{stdaz2})^2$$

$$\text{sigrzs}_i = (\text{stdel1})^2 + (|Y_i| \text{stdel2})^2.$$

sigmr, stdaz1, stdaz2, stdel1, stdel2, are constants set by the user. Nominal values are 5(meters), 1(meter), .0005 (mils ÷ 1000), 1(meter), .0005(mils ÷ 1000), respectively.

We assume that the tracking radar which generated the Z_i was well calibrated and therefore had zero bias. In other words,

$$\bar{V}_i = 0 \quad i=1 \dots n.$$

3.3 The Target Dynamical Inputs ϕ , Q , Γ , and \bar{W} .

ϕ_i represents the dynamics of an aircraft which is controlled at time step i through its acceleration rate, $\Gamma_i W_i$. Alternatively, we can look on ϕ_i as representing an aircraft whose acceleration is constant over any particular transition, $i \rightarrow i+1$, where the error in this representation is given by $\Gamma_i W_i$.

$$\phi_i = \begin{pmatrix} I_{3 \times 3} & I_{3 \times 3} \Delta t & .5 I_{3 \times 3} \Delta t^2 \\ 0_{3 \times 3} & I_{3 \times 3} & I_{3 \times 3} \Delta t \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \end{pmatrix}$$

where Δt = time step between iterations. Note that ϕ is constant over all i .

The computations of the inputs \bar{W}_i and Q_i are complicated in that they are physically related to the aircraft body frame variables of thrust rate, g rate, and roll rate and their corresponding covariances. The transition from body frame dynamics to reference frame dynamics is worked out in Reference 5. At any particular instant, a circle is matched to the target's flight path with the properties that the component of target acceleration perpendicular to the velocity vector points towards the circle's center. The target's body frame has its first component axis along the velocity vector and its second component axis along the normal acceleration vector. It is also called the Frenet frame. Γ is the body, or Frenet frame, to reference frame coordinate transformation. It is computed at each i based on the latest available state estimate. Γ 's matrix components are;

$$11_{\Gamma} = 4_Y \div v$$

$$21_{\Gamma} = 5_Y \div v$$

$$31_{\Gamma} = 6_Y \div v$$

$$12_{\Gamma} = (7_Y - 4_Y at) \div gn$$

$$22_{\Gamma} = (8_Y - 5_Y at) \div gn$$

$$32_{\Gamma} = (9_Y - 6_Y at) \div gn$$

$$13_{\Gamma} = 21_{\Gamma} 32_{\Gamma} - 31_{\Gamma} 22_{\Gamma}$$

$$23_{\Gamma} = 12_{\Gamma} 31_{\Gamma} - 11_{\Gamma} 32_{\Gamma}$$

$$33_{\Gamma} = 11_{\Gamma} 22_{\Gamma} - 12_{\Gamma} 21_{\Gamma}$$

where

$$v = (4_Y^2 + 5_Y^2 + 6_Y^2)^{1/2}$$

$$k = [({}^5_Y{}^9_Y - {}^6_Y{}^8_Y)^2 + ({}^4_Y{}^9_Y - {}^7_Y{}^6_Y)^2 + ({}^4_Y{}^8_Y - {}^5_Y{}^7_Y)^2]^{1/2} \div v^3$$

$$at = ({}^4_Y{}^7_Y + {}^5_Y{}^8_Y - {}^6_Y{}^9_Y) \div v^2$$

$$gn = kv^2.$$

The computation of \bar{W} is based on the assumption that the aircraft flies along the Frenet circle over time Δt .

$$\bar{W} = \begin{pmatrix} -k^2 v^3 \Delta t^3 \div 6 \\ kvv \Delta t^3 \div 6 \\ 0 \\ -k^2 v^3 \Delta t^2 \div 2 \\ kvv \Delta t^2 \div 2 \\ 0 \\ -k^2 v^3 \Delta t \\ kvv \Delta t \\ 0 \end{pmatrix}$$

where

$$\dot{v} = ({}^4_Y{}^7_Y + {}^4_Y{}^8_Y + {}^6_Y{}^9_Y) \div v.$$

The corresponding transition error covariance is

$$Q = \begin{pmatrix} QAP \Delta t^4 \div 36 & QAP \Delta t^3 \div 12 & QAP \Delta t^2 \div 2 \\ QAP \Delta t^3 \div 12 & QAP \Delta t^2 \div 2 & QAP \Delta t \\ QAP \Delta t^2 \div 2 & QAP \Delta t & QAP \end{pmatrix}$$

where

$$QAP = \begin{pmatrix} (C_1 \Delta t)^2 & 0 & 0 \\ 0 & (C_2 \Delta t)^2 & 0 \\ 0 & 0 & (g n * C_3 \Delta t)^2 \end{pmatrix} .$$

C_1 , C_2 , C_3 are constants set by the user corresponding to the one standard deviation (STD) of thrust rate, g rate, and roll rate, respectively. Nominal values are 1.0(meters/sec³), 7.0(meters/sec³), and 1.0(radians/sec respectively. Note, for notational convenience the i subscripts have been omitted in all the above equations.

3.4 Remarks on Nonlinearities.

The input variables can in principle be defined in a variety of ways. For example, they could each be referred to any number of possibly different coordinate systems. Unfortunately, the nature of the tracking problem will result in any, but the most simple minded, formulation having nonlinear portions. The investigator can influence where these nonlinearities occur. In our formulation they have been shifted to the coordinate transition matrices TRS and Γ .

Nonlinearities have two effects. On the forward sweep, Γ and TRS depend on the most recent state estimates, Y_i , for their computation. Therefore the P_i cannot be a priori computed but must be determined simultaneously with the Y_i . Secondly, we must change our interpretation of the δ operator in equations (73) through (108) from that of variation to that of small variation.

The nonlinearities raise some question as to the adequacy of the \bar{W} and Q computations, particularly for severely jinking targets where lags in the forward sweep estimates may become large. Conceivably we can reduce the lags by increasing Q , that is increasing C_1 , C_2 , C_3 , and/or running the smoother back and forth over the data several times, each time starting with \bar{X}_0 and \bar{P}_0 from the previous

backward sweep. Doubtless the reader can think of other schemes for reducing the effects of nonlinearities.

Next page is blank

4. USERS' GUIDE

The program uses Top Down Structured Programming (TDSP) and is written in simple, machine independent FORTRAN. In line with TDSP, the coding uses only five logical structures, is self-documented (requiring no flow charts) and is completely straightforward (avoiding "programming tricks") (Reference 6). A brief description of the program's overall structure and a few remarks on input data are all a perspective user will need for a guide.

4.1 Overall Structure.

A schematic diagram of the functional hierarchy is given in Figure 4.1.

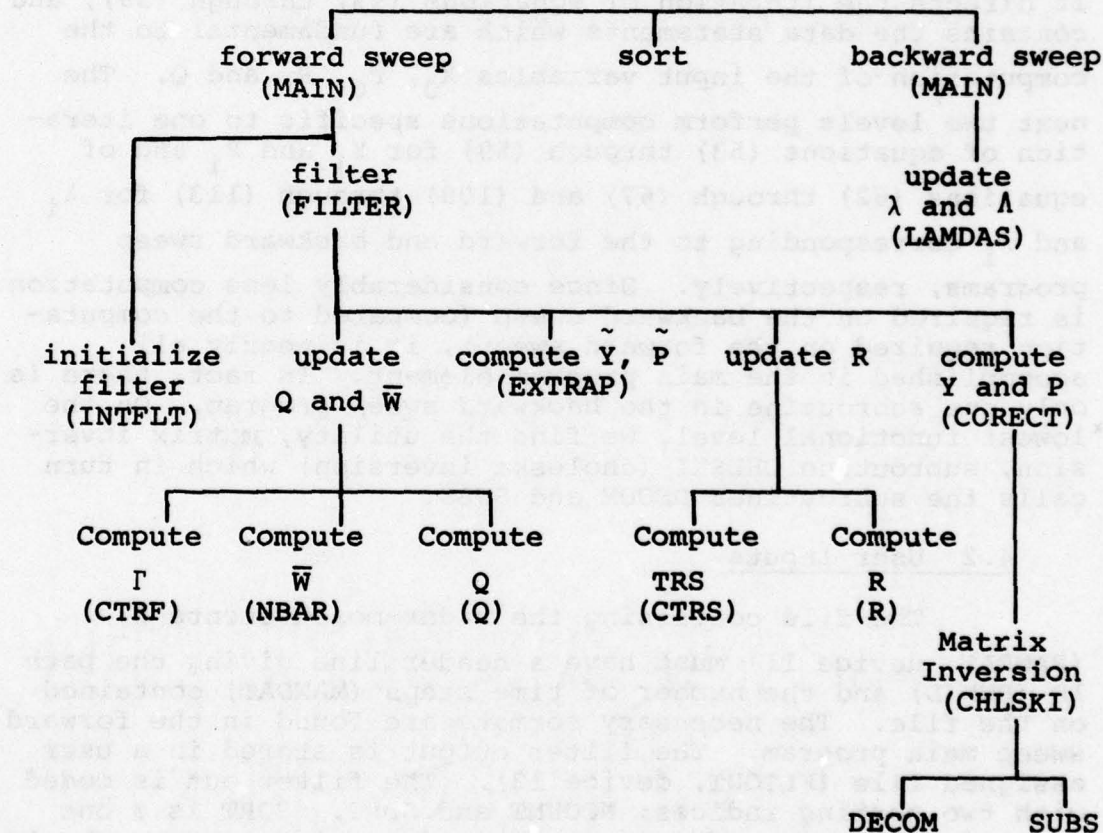


Figure 4.1 Programming Structure (Subroutine Name)

The first level consists of three independent programs, the forward sweep, sort, and backward sweep programs. The forward sweep reads the tracking data from a user selected storage device (e.g., FASTRAND, tape, disc), filters it, and writes the results on user selected devices. The sort program reverses the chronological order of the filtered output in preparation for the backward sweep. The user is urged to consult his own computer facility for a machine optimized sort routine. For example, AMSAA's Air Defense Evaluation Branch uses a sort program optimized for the Univac 1108 consisting of one executive command, @MISD* L1B\$.SORTSDF 12., 12.,#records, #chars/per, KEY/127/4.D, KEY/131/2.A. The backward sweep reads the filtered output in reverse chronological order and writes the final smoothed results on user selected devices. The user may wish to use SORT again since the smoothed results are in reverse chronological order.

The next functional level is the FILTER subroutine. It directs the iteration of equations (53) through (59), and contains the data statements which are fundamental to the computation of the input variables \bar{X}_0 , \bar{P}_0 , R, and Q. The next two levels perform computations specific to one iteration of equations (53) through (59) for Y_i and P_i and of equations (62) through (67) and (109) through (113) for λ_i and Λ_i corresponding to the forward and backward sweep programs, respectively. Since considerably less computation is required on the backward sweep (compared to the computation required on the forward sweep), it is nearly all accomplished in the main program element. In fact, there is only one subroutine in the backward sweep program. On the lowest functional level, we find the utility, matrix inversion, subroutine CHLSKI (choleski inversion) which in turn calls the subroutines DECOM and SUBS.

4.2 User Inputs.

The file containing the radar measurements Z_i (RAWDAT, device 11) must have a header line giving the path ID (RUNID) and the number of time steps (MAXDAT) contained on the file. The necessary formats are found in the forward sweep main program. The filter output is stored in a user assigned file (FLTOUT, device 12). The filter out is coded with two sorting indices; NCOUNT and SORT. SORT is a one dimensional array. The user selected sorting routine should sort in decreasing numerical order on NCOUNT and in normal ascending order on SORT. The backward sweep program reads the sorted (reverse chronological order) data from FLTOUT and writes the smoothed data in a user assigned file (SMTOUT, device 13).

The data time step, DELTAT, must be set in each of the two main program elements. Subroutine INTFLT contains all the remaining variables that are to be set by the user. They are;

CONE, the STD of target thrust rate,
 CTWO, the STD of target g rate,
 CTHREE, the STD of target roll rate,
 SIGMR, the STD of range measurement error,
 STDAZ1, the STD of linear azimuth measurement error,
 STDAZ2, the STD of angular azimuth measurement error,
 STDEL1, the STD of linear elevation measurement error,
 STDEL2, the STD of angular elevation measurement error,
 PVAL(i), i = 1...9, are the STDs of the initial state vector in the X, Y, Z directions of position, velocity, and acceleration, respectively.

4.3 Identification of Program Names.

This section identifies the FORTRAN names of the principal variables used in the program with the corresponding symbols used in the theory of sections 2 and 3.

<u>Program Name</u>	<u>Theory Name</u>
BARN	$\Gamma \bar{W}$
CAPLAM	Λ
DELTAT	Δt
HKAL	H
LAMDA	λ
PKAL	P, \tilde{P}
QA	$\Gamma QAP \Gamma^T$
QAP	QAP
QKAL	$\Gamma Q \Gamma^T$

<u>Program Name</u>	<u>Theory Name</u>
RESDUL	Z-HY
RMEAS	R
RMEASI	R^{-1}
SAB	A
SMTBRN	$\Gamma \hat{W}$
SMTTP	SP
SMTQ	$\Gamma SQ \Gamma^T$
SMTSPS	\hat{X}
SPS	Y, \tilde{Y}
TRS	TRS
TRV	Γ^T
ZMEAS	Z

4.3 Identification of Program Names

This section identifies the FORTRAN names of the principal variables used in the program with the corresponding symbols used in the theory of sections 2 and 3.

<u>Theory Name</u>	<u>FORTRAN Name</u>
BARN	
CAPLAN	
EDTAT	
HEAL	
LANTA	
PRAL	
QA	
QAL	
QAL	
QAL	

REFERENCES

1. Meredith, J., Schöder, R., Lufkin, B., Evaluation of the Gun Low Altitude Fire Control Test Bed (GLAADS), AMSAA Technical Report No. 149, April 1977.
2. Bryson, A.E., Jr., and Yu-Chi Ho, Applied Optimal Control, Waltham, MA, Ginn and Company, 1969.
3. Canon, M.D., Clifton, D.C., Polak, E., Theory of Optimal Control and Mathematical Programming, New York, NY, McGraw-Hill, 1970.
4. Bryson, A.E., and Frazier, M., Smoothing for Linear and Nonlinear Dynamic Systems, Proceedings of the Optimum System Synthesis Conference, Wright-Patterson Air Force Base, Ohio, September 1963.
5. Scheder, R.A., Adaptive Esitnation, AMSAA Technical Report No. 166, December 1976.
6. Mills, H.D., Mathematical Foundations for Structured Programming, IBM Corporation, Federal Systems Division, Gaithersburg, MD, 1972.

Next page is blank

Page	Index	Table
41	Forward Sweep Main Program	A-1
42	Backward Sweep Main Program	A-2
43	Subroutine CHIRP	A-3
44	Subroutine CORRECT	A-4
45	Subroutine CTR1	A-5
46	Subroutine CTR2	A-6
47	Subroutine DECOM	A-7
48	Subroutine EXTRAL	A-8
49	Subroutine FINDER	A-9
50	Subroutine INITIAT	A-10
51	Subroutine LAMDA	A-11
52	Subroutine NBAR	A-12
53	Subroutine Q	A-13
54	Subroutine R	A-14
55	Subroutine SUBS	A-15

APPENDIX A
PROGRAM LISTING

INDEX

<u>Table</u>	<u>Title</u>	<u>Page</u>
A-1	Forward Sweep Main Program	41
A-2	Backward Sweep Main Program	44
A-3	Subroutine CHLSKI	49
A-4	Subroutine CORECT	50
A-5	Subroutine CTRF	52
A-6	Subroutine CTRS	55
A-7	Subroutine DECOM	57
A-8	Subroutine EXTRAP	58
A-9	Subroutine FILTER	60
A-10	Subroutine INTFLT	61
A-11	Subroutine LAMDAS	64
A-12	Subroutine NBAR	67
A-13	Subroutine Q	69
A-14	Subroutine R	71
A-15	Subroutine SUBS	73

Table A-1 Forward Sweep Main Program

1	C	THIS PROGRAM ESTIMATES TARGET POSITION, VELOCITY, AND ACCELERATION
2	C	BY SWEEPING THRU RAW TRACKING (POSITION) DATA USING A SELF
3	C	ADAPTING, NON-LINEAR, KALMAN FILTER.
4	C	
5	C	REFERENCES
6	C	
7	C	1. SCHEDER R.A., ADAPTIVE ESTIMATION,
8	C	AMSAA TECHNICAL REPORT 166
9	C	APG, MD 1975.
10	C	
11	C	2. BRYSON, A.E. JR., AND HO YU-CHI,
12	C	APPLIED OPTIMAL CONTROL, GINN AND COMPANY,
13	C	WALTHAM, MA 1969, CHAPTER 12
14	C	
15	C	COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,QKAL
16	C	,ZNEAS ,HKAL ,PHI ,RMEAS
17	C	
18	C	COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE,SIGMR
19	C	,STDAZ1,STDAZ2,STDEL1,STDEL2
20	C	
21	C	REAL RESDUL(3)
22	C	,BARN(9)
23	C	,SPS(9)
24	C	,PKAL(9,9)
25	C	,QA(3,3)
26	C	,RMEASI(3,3)
27	C	,QKALI(9,9)
28	C	,ZNEAS(3)
29	C	,HKAL(3,9)
30	C	,PHI(9,9)
31	C	,RMEAS(3,3)
32	C	
33	C	REAL PVAL(9)
34	C	,DELTAT
35	C	,CONE
36	C	,CTWO
37	C	,CTHREE
38	C	,SIGMR
39	C	,STDAZ1
40	C	,STDAZ2
41	C	,STDEL1
42	C	,STDEL2
43	C	
44	C	REAL ADOT(3)
45	C	
46	C	INTEGER RAWDAT
47	C	,FLYOUT
48	C	,RUNID
49	C	,MAXDAT
50	C	,SORT(20)
51	C	
52	C	FORMAT (4F10.2)
53	C	FORMAT (110,F10.2,6F14.7,22X,14,12,/, (9F14.7,14,12))
54	C	FORMAT (1X,14,F10.2,9F9.1,2X,3F9.2)
55	C	FORMAT (1X, 'STATE ERROR COVARIANCE', 9(/,1X,9F10.2,
56	C	/,1X, 'ACCELERATION TRANSITION ERROR COVARIANCE',

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

ENDFILE FLTOUT

STOP
END

C

114
115
116
117

C THIS PROGRAM SMOOTHS TRACKING DATA BY SWEEPING BACKWARDS
C THROUGH PREVIOUSLY FILTERED DATA.

REFERENCES

1. MEREDITH J., SCHEDER R., LUFKIN B., EVALUATION OF THE GUN LOW ALTITUDE AIR DEFENSE FIRE CONTROL TEST BED (GLAD), AMSAA TECHNICAL REPORT 149, APR. MD 1977. CHAPTER 3

2. BRYSON, A.E., JR., AND HO, YU-CHI,
APPLIED OPTIMAL CONTROL, GINN AND COMPANY,
WALTHAM, MA 1969. CHAPTER 13

COMMON /KALIO /	RESUL,BARN	,SPS	,PKAL	,QA
	,ZMEAS',HKAL	,PHI	,RMEAS	,RMEASI,QKAL
A				

COMMON /ABIO / LAMDA / CAPLAM

DOUBLE PRECISION LAMDA(9)
CAPLAM(9,9)

DOUBLE PRECISION RESDUL(3)

A
BARN(3)

B .SPS(9)

C .PKAL(9,9)

.QA(3.3)

.RMEASI(3,3)

OKAL(9,9)

MEAS (3)

HKAL(3,9)
DWT(9,9)

PHI(9,9)
PMEAS(3,3)

0, KNEAS (3.3)

DOUBLE PRECISION TEMP3(9,9)

A .SMTSPS(9)

B .SMTBRN(9)

TEMP4(9,9)

E . SMTP(9.9)

F ,SMTQ(9,9)

2000

REAL ADOT(3)

INTEGERS

INTEGER FL100T
CONT/20

SMITH
SMITH

PRINTED
IN U.S.A.

MAXDAT

INDEX

FORMAT (10X.F10)

FORMAT (1X, I4, F

FORMAT (1X, 'STA

 $\cdot .1x. 'AC$

B .36/.1X.

FORMAT (I10,F10)

FORMAT (I10)

© 2000 Blackwell Science Ltd *Journal of Internal Medicine* 247: 399–405

1

44

```

57 700 FORMAT (I10,I16X,I4,I2)
58 C
59 C INITIALIZE SMOOTHER
60 C
61 DELTAT = 0.05
62 FLTOUT = 12
63 SMTOUT = 13
64 C
65 REWIND SMTOUT
66 REWIND FLTOUT
67 C
68 READ (FLTOUT,700) RUNID ,MAXDAT, IC
69 WRITE (SMTOUT,700) RUNID ,MAXDAT, IC
70 WRITE (6,600) RUNID
71 C
72 DO 0950 I = 1,20
73 SORT(I) = I
74 CONTINUE
75 C
76 STATE TO MEASUREMENT TRANSITION MATRIX HKAL
77 C
78 DO 1500 I = 1,3
79 DO 1000 J = 1,9
80 HKAL(I,J) = 0.0
81 CONTINUE
82 CONTINUE
83 HKAL(1,1) = 1.0
84 HKAL(2,2) = 1.0
85 HKAL(3,3) = 1.0
86 C
87 STATE TRANSITION MATRIX PHI
88 C
89 DO 3000 I = 1,9
90 DO 2000 J = 1,9
91 PHI(I,J) = 0.0
92 CONTINUE
93 CONTINUE
94 DO 4000 I = 1,3
95 PHI(I,1) = 1.0
96 PHI(I+3,1+3) = 1.0
97 PHI(I+6,1+6) = 1.0
98 CONTINUE
99 C
100 PHI(1,4) = DELTAT
101 PHI(2,5) = DELTAT
102 PHI(3,6) = DELTAT
103 PHI(1,7) = DELTAT**2 / 2.0
104 PHI(2,8) = DELTAT**2 / 2.0
105 PHI(3,9) = DELTAT**2 / 2.0
106 PHI(4,7) = DELTAT
107 PHI(5,8) = DELTAT
108 PHI(6,9) = DELTAT
109 C
110 LAMDA AND CAPLAM
111 C
112 DO 6000 I = 1,9
113 LAMDA(I) = 0.0

```

```

114 DO 5000 J = 1.9
115 CAPLAM(I,J) = 0.0
116 CONTINUE
117 CONTINUE
118
119 BACKWARD SWEEP
120 C
121 C
122 C
123 C
124 C
125 C
126
127 INPUT
128 ( ( FILTERD DATA MUST BE IN REVERSE CHRONOLOGICAL ORDER )
129 READ(FLTOUT,200) TIME,(RESUL(I),I=1,3)
130 .(BARN(I),I=7,9)
131 .(SPS(I),I=1,9)
132 .((PKAL(I,J),J=1,9),I=1,9)
133 .((QA(I,J),J=1,3),I=1,3)
134 .((RMEASI(I,J),J=1,3),I=1,3)
135
136 EXPAND BARN AND QA TO FULL DIMENSION
137
138 DO 7000 I = 1.3
139 BARN(I) = 0.1666667*BARN(I+6)*DELTAT**2
140 BARN(I+3) = 0.5*BARN(I+6)*DELTAT
141 CONTINUE
142
143 DO 7400 I = 1.3
144 DO 7300 J = 1.3
145 QKAL(I,J) = QA(I,J)*DELTAT**4/36.
146 QKAL(I,J+3) = QA(I,J)*DELTAT**3/12.
147 QKAL(I,J+6) = QA(I,J) * DELTAT**2 / 6.0
148 QKAL(I+3,J) = QA(I,J)*DELTAT**3/12.
149 QKAL(I+3,J+3) = QA(I,J) * DELTAT**2 / 4.0
150 QKAL(I+3,J+6) = QA(I,J) * DELTAT / 2.0
151 QKAL(I+6,J) = QA(I,J) * DELTAT**2 / 6.0
152 QKAL(I+6,J+3) = QA(I,J) * DELTAT / 2.0
153 QKAL(I+6,J+6) = QA(I,J)
154 CONTINUE
155 CONTINUE
156
157 COMPUTE SMOOTHED STATE
158 PKAL * PHI(TRANPOSE) STORE IN TEMP3
159
160 DO 9000 I = 1.9
161 DO 8000 J = 1.9
162 TEMP3(I,J) = 0.0
163 DO 7500 K = 1.9
164 TEMP3(I,J) = TEMP3(I,J) + PKAL(I,K) * PHI(J,K)
165 CONTINUE
166 CONTINUE
167 CONTINUE
168
169 SMTSPS = SPS - TEMP3 * LAMDA --- SMOOTHED STATE
170 SMTBRN = BARN - QFLT * LAMDA ---SMOOTHED TRANSITION ERROR
171 DO 10000 I = 1.9

```

```

171 SMTSPS(I) = SPS(I)
172 SMTBRN(I) = BARN(I)
173 DO 9500 J = 1,9
174   SMTSPS(I) = SMTSPS(I) - TEMP3(I,J)*LAMDA(J)
175   SMTBRN(I) = SMTBRN(I) - QKAL(I,J) * LAMDA(J)
176   CONTINUE
177   9500
10000
178 C
179 C
180 C
181 C
182 C
183 C
184 C
185 C
186 C
187 C
188 C
189 C
190 C
191 C
192 C
193 C
194 C
195 C
196 C
197 C
198 C
199 C
200 C
201 C
202 C
203 C
204 C
205 C
206 C
207 C
208 C
209 C
210 C
211 C
212 C
213 C
214 C
215 C
216 C
217 C
218 C
219 C
220 C
221 C
222 C
223 C
224 C
225 C
226 C
227 C

      COMPUTE SMOOTHED COVARIANCES
      TEMP3 * CAPLAM * TEMP3(TRANPOSE)      STORE IN TEMP4

      DO 1'000 I = 1,9
      DO 10500 J = 1,9
      TEMP4(I,J) = 0.0
      DO 10200 M = 1,9
      DO 10100 K = 1,9
      TEMP4(I,J) = TEMP4(I,J) + TEMP3(I,M) * CAPLAM(M,K) *
      TEMP3(J,K)
      CONTINUE
      CONTINUE
      CONTINUE
      CONTINUE
      QKAL * CAPLAM * QKAL      STORE IN TEMP3

      DO 12000 I = 1,9
      DO 11500 J = 1,9
      TEMP3(I,J) = 0.0
      DO 11200 M = 1,9
      DO 11100 K = 1,9
      TEMP3(I,J) = TEMP3(I,J) + QKAL(I,M) * CAPLAM(M,K) *
      QKAL(K,J)
      CONTINUE
      CONTINUE
      CONTINUE
      CONTINUE
      SMTP = PKAL - TEMP4      --- SMOOTHED STATE ERROR COVARIANCE
      SMTQ = QKAL - TEMP3      --- SMOOTHED PLANT ERROR COVARIANCE

      DO 13000 I = 1,9
      DO 12500 J = 1,9
      SMTP(I,J) = PKAL(I,J) - TEMP4(I,J)
      SMTQ(I,J) = QKAL(I,J) - TEMP3(I,J)
      CONTINUE
      CONTINUE
      12500
13000
227 C
228 C
229 C
230 C
231 C
232 C
233 C
234 C
235 C
236 C
237 C
238 C
239 C
240 C
241 C
242 C
243 C
244 C
245 C
246 C
247 C
248 C
249 C
250 C
251 C
252 C
253 C
254 C
255 C
256 C
257 C
258 C
259 C
260 C
261 C
262 C
263 C
264 C
265 C
266 C
267 C
268 C
269 C
270 C
271 C
272 C
273 C
274 C
275 C
276 C
277 C
278 C
279 C
280 C
281 C
282 C
283 C
284 C
285 C
286 C
287 C
288 C
289 C
290 C
291 C
292 C
293 C
294 C
295 C
296 C
297 C
298 C
299 C
300 C
301 C
302 C
303 C
304 C
305 C
306 C
307 C
308 C
309 C
310 C
311 C
312 C
313 C
314 C
315 C
316 C
317 C
318 C
319 C
320 C
321 C
322 C
323 C
324 C
325 C
326 C
327 C
328 C
329 C
330 C
331 C
332 C
333 C
334 C
335 C
336 C
337 C
338 C
339 C
340 C
341 C
342 C
343 C
344 C
345 C
346 C
347 C
348 C
349 C
350 C
351 C
352 C
353 C
354 C
355 C
356 C
357 C
358 C
359 C
360 C
361 C
362 C
363 C
364 C
365 C
366 C
367 C
368 C
369 C
370 C
371 C
372 C
373 C
374 C
375 C
376 C
377 C
378 C
379 C
380 C
381 C
382 C
383 C
384 C
385 C
386 C
387 C
388 C
389 C
390 C
391 C
392 C
393 C
394 C
395 C
396 C
397 C
398 C
399 C
400 C
401 C
402 C
403 C
404 C
405 C
406 C
407 C
408 C
409 C
410 C
411 C
412 C
413 C
414 C
415 C
416 C
417 C
418 C
419 C
420 C
421 C
422 C
423 C
424 C
425 C
426 C
427 C
428 C
429 C
430 C
431 C
432 C
433 C
434 C
435 C
436 C
437 C
438 C
439 C
440 C
441 C
442 C
443 C
444 C
445 C
446 C
447 C
448 C
449 C
450 C
451 C
452 C
453 C
454 C
455 C
456 C
457 C
458 C
459 C
460 C
461 C
462 C
463 C
464 C
465 C
466 C
467 C
468 C
469 C
470 C
471 C
472 C
473 C
474 C
475 C
476 C
477 C
478 C
479 C
480 C
481 C
482 C
483 C
484 C
485 C
486 C
487 C
488 C
489 C
490 C
491 C
492 C
493 C
494 C
495 C
496 C
497 C
498 C
499 C
500 C
501 C
502 C
503 C
504 C
505 C
506 C
507 C
508 C
509 C
510 C
511 C
512 C
513 C
514 C
515 C
516 C
517 C
518 C
519 C
520 C
521 C
522 C
523 C
524 C
525 C
526 C
527 C
528 C
529 C
530 C
531 C
532 C
533 C
534 C
535 C
536 C
537 C
538 C
539 C
540 C
541 C
542 C
543 C
544 C
545 C
546 C
547 C
548 C
549 C
550 C
551 C
552 C
553 C
554 C
555 C
556 C
557 C
558 C
559 C
560 C
561 C
562 C
563 C
564 C
565 C
566 C
567 C
568 C
569 C
570 C
571 C
572 C
573 C
574 C
575 C
576 C
577 C
578 C
579 C
580 C
581 C
582 C
583 C
584 C
585 C
586 C
587 C
588 C
589 C
590 C
591 C
592 C
593 C
594 C
595 C
596 C
597 C
598 C
599 C
600 C
601 C
602 C
603 C
604 C
605 C
606 C
607 C
608 C
609 C
610 C
611 C
612 C
613 C
614 C
615 C
616 C
617 C
618 C
619 C
620 C
621 C
622 C
623 C
624 C
625 C
626 C
627 C
628 C
629 C
630 C
631 C
632 C
633 C
634 C
635 C
636 C
637 C
638 C
639 C
640 C
641 C
642 C
643 C
644 C
645 C
646 C
647 C
648 C
649 C
650 C
651 C
652 C
653 C
654 C
655 C
656 C
657 C
658 C
659 C
660 C
661 C
662 C
663 C
664 C
665 C
666 C
667 C
668 C
669 C
670 C
671 C
672 C
673 C
674 C
675 C
676 C
677 C
678 C
679 C
680 C
681 C
682 C
683 C
684 C
685 C
686 C
687 C
688 C
689 C
690 C
691 C
692 C
693 C
694 C
695 C
696 C
697 C
698 C
699 C
700 C
701 C
702 C
703 C
704 C
705 C
706 C
707 C
708 C
709 C
710 C
711 C
712 C
713 C
714 C
715 C
716 C
717 C
718 C
719 C
720 C
721 C
722 C
723 C
724 C
725 C
726 C
727 C
728 C
729 C
730 C
731 C
732 C
733 C
734 C
735 C
736 C
737 C
738 C
739 C
740 C
741 C
742 C
743 C
744 C
745 C
746 C
747 C
748 C
749 C
750 C
751 C
752 C
753 C
754 C
755 C
756 C
757 C
758 C
759 C
760 C
761 C
762 C
763 C
764 C
765 C
766 C
767 C
768 C
769 C
770 C
771 C
772 C
773 C
774 C
775 C
776 C
777 C
778 C
779 C
780 C
781 C
782 C
783 C
784 C
785 C
786 C
787 C
788 C
789 C
790 C
791 C
792 C
793 C
794 C
795 C
796 C
797 C
798 C
799 C
800 C
801 C
802 C
803 C
804 C
805 C
806 C
807 C
808 C
809 C
810 C
811 C
812 C
813 C
814 C
815 C
816 C
817 C
818 C
819 C
820 C
821 C
822 C
823 C
824 C
825 C
826 C
827 C
828 C
829 C
830 C
831 C
832 C
833 C
834 C
835 C
836 C
837 C
838 C
839 C
840 C
841 C
842 C
843 C
844 C
845 C
846 C
847 C
848 C
849 C
850 C
851 C
852 C
853 C
854 C
855 C
856 C
857 C
858 C
859 C
860 C
861 C
862 C
863 C
864 C
865 C
866 C
867 C
868 C
869 C
870 C
871 C
872 C
873 C
874 C
875 C
876 C
877 C
878 C
879 C
880 C
881 C
882 C
883 C

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

228      ADOT(I) = SMTBRN(I+6) / DELTAT
229      CONTINUE
230
231      WRITE (6,300) NCOUNT, TIME, (SMTSPS(I), I=1,9), (ADOT(I), I=1,3)
232
233      IF (MOD((NCOUNT - 1), 20) .NE. 0) GO TO 14000
234      WRITE (6,400) ((SMTSP(I,J), J=1,9), I=1,9)
235      WRITE (6,400) ((SMTQ(I,J), J=7,9), I=7,9)
236      CONTINUE
237
238      A
239      14000
240      C
241      C
242      C
243      C
244      C
245      C
246      C
247      C
248      C
249      C
250      C
251      C
252      C
253      C
254      C
255      C
256      C
257      C
258      C
259      C
260      C
261      C
262      C
263      C
264      C
265      C
266      C
267      C
268      C
269      C
270      C
271      C
272      C
273      C
274      C
275      C
276      C
277      C
278      C
279      C
280      C
281      C
282      C
283      C
284      C
285      C
286      C
287      C
288      C
289      C
290      C
291      C
292      C
293      C
294      C
295      C
296      C
297      C
298      C
299      C
300      C
301      C
302      C
303      C
304      C
305      C
306      C
307      C
308      C
309      C
310      C
311      C
312      C
313      C
314      C
315      C
316      C
317      C
318      C
319      C
320      C
321      C
322      C
323      C
324      C
325      C
326      C
327      C
328      C
329      C
330      C
331      C
332      C
333      C
334      C
335      C
336      C
337      C
338      C
339      C
340      C
341      C
342      C
343      C
344      C
345      C
346      C
347      C
348      C
349      C
350      C
351      C
352      C
353      C
354      C
355      C
356      C
357      C
358      C
359      C
360      C
361      C
362      C
363      C
364      C
365      C
366      C
367      C
368      C
369      C
370      C
371      C
372      C
373      C
374      C
375      C
376      C
377      C
378      C
379      C
380      C
381      C
382      C
383      C
384      C
385      C
386      C
387      C
388      C
389      C
390      C
391      C
392      C
393      C
394      C
395      C
396      C
397      C
398      C
399      C
400      C
401      C
402      C
403      C
404      C
405      C
406      C
407      C
408      C
409      C
410      C
411      C
412      C
413      C
414      C
415      C
416      C
417      C
418      C
419      C
420      C
421      C
422      C
423      C
424      C
425      C
426      C
427      C
428      C
429      C
430      C
431      C
432      C
433      C
434      C
435      C
436      C
437      C
438      C
439      C
440      C
441      C
442      C
443      C
444      C
445      C
446      C
447      C
448      C
449      C
450      C
451      C
452      C
453      C
454      C
455      C
456      C
457      C
458      C
459      C
460      C
461      C
462      C
463      C
464      C
465      C
466      C
467      C
468      C
469      C
470      C
471      C
472      C
473      C
474      C
475      C
476      C
477      C
478      C
479      C
480      C
481      C
482      C
483      C
484      C
485      C
486      C
487      C
488      C
489      C
490      C
491      C
492      C
493      C
494      C
495      C
496      C
497      C
498      C
499      C
500      C
501      C
502      C
503      C
504      C
505      C
506      C
507      C
508      C
509      C
510      C
511      C
512      C
513      C
514      C
515      C
516      C
517      C
518      C
519      C
520      C
521      C
522      C
523      C
524      C
525      C
526      C
527      C
528      C
529      C
530      C
531      C
532      C
533      C
534      C
535      C
536      C
537      C
538      C
539      C
540      C
541      C
542      C
543      C
544      C
545      C
546      C
547      C
548      C
549      C
550      C
551      C
552      C
553      C
554      C
555      C
556      C
557      C
558      C
559      C
560      C
561      C
562      C
563      C
564      C
565      C
566      C
567      C
568      C
569      C
570      C
571      C
572      C
573      C
574      C
575      C
576      C
577      C
578      C
579      C
580      C
581      C
582      C
583      C
584      C
585      C
586      C
587      C
588      C
589      C
590      C
591      C
592      C
593      C
594      C
595      C
596      C
597      C
598      C
599      C
600      C
601      C
602      C
603      C
604      C
605      C
606      C
607      C
608      C
609      C
610      C
611      C
612      C
613      C
614      C
615      C
616      C
617      C
618      C
619      C
620      C
621      C
622      C
623      C
624      C
625      C
626      C
627      C
628      C
629      C
630      C
631      C
632      C
633      C
634      C
635      C
636      C
637      C
638      C
639      C
640      C
641      C
642      C
643      C
644      C
645      C
646      C
647      C
648      C
649      C
650      C
651      C
652      C
653      C
654      C
655      C
656      C
657      C
658      C
659      C
660      C
661      C
662      C
663      C
664      C
665      C
666      C
667      C
668      C
669      C
670      C
671      C
672      C
673      C
674      C
675      C
676      C
677      C
678      C
679      C
680      C
681      C
682      C
683      C
684      C
685      C
686      C
687      C
688      C
689      C
690      C
691      C
692      C
693      C
694      C
695      C
696      C
697      C
698      C
699      C
700      C
701      C
702      C
703      C
704      C
705      C
706      C
707      C
708      C
709      C
710      C
711      C
712      C
713      C
714      C
715      C
716      C
717      C
718      C
719      C
720      C
721      C
722      C
723      C
724      C
725      C
726      C
727      C
728      C
729      C
730      C
731      C
732      C
733      C
734      C
735      C
736      C
737      C
738      C
739      C
740      C
741      C
742      C
743      C
744      C
745      C
746      C
747      C
748      C
749      C
750      C
751      C
752      C
753      C
754      C
755      C
756      C
757      C
758      C
759      C
760      C
761      C
762      C
763      C
764      C
765      C
766      C
767      C
768      C
769      C
770      C
771      C
772      C
773      C
774      C
775      C
776      C
777      C
778      C
779      C
780      C
781      C
782      C
783      C
784      C
785      C
786      C
787      C
788      C
789      C
790      C
791      C
792      C
793      C
794      C
795      C
796      C
797      C
798      C
799      C
800      C
801      C
802      C
803      C
804      C
805      C
806      C
807      C
808      C
809      C
810      C
811      C
812      C
813      C
814      C
815      C
816      C
817      C
818      C
819      C
820      C
821      C
822      C
823      C
824      C
825      C
826      C
827      C
828      C
829      C
830      C
831      C
832      C
833      C
834      C
835      C
836      C
837      C
838      C
839      C
840      C
841      C
842      C
843      C
844      C
845      C
846      C
847      C
848      C
849      C
850      C
851      C
852      C
853      C
854      C
855      C
856      C
857      C
858      C
859      C
860      C
861      C
862      C
863      C
864      C
865      C
866      C
867      C
868      C
869      C
870      C
871      C
872      C
873      C
874      C
875      C
876      C
877      C
878      C
879      C
880      C
881      C
882      C
883      C
884      C
885      C
886      C
887      C
888      C
889      C
890      C
891      C
892      C
893      C
894      C
895      C
896      C
897      C
898      C
899      C
900      C
901      C
902      C
903      C
904      C
905      C
906      C
907      C
908      C
909      C
910      C
911      C
912      C
913      C
914      C
915      C
916      C
917      C
918      C
919      C
920      C
921      C
922      C
923      C
924      C
925      C
926      C
927      C
928      C
929      C
930      C
931      C
932      C
933      C
934      C
935      C
936      C
937      C
938      C
939      C
940      C
941      C
942      C
943      C
944      C
945      C
946      C
947      C
948      C
949      C
950      C
951      C
952      C
953      C
954      C
955      C
956      C
957      C
958      C
959      C
960      C
961      C
962      C
963      C
964      C
965      C
966      C
967      C
968      C
969      C
970      C
971      C
972      C
973      C
974      C
975      C
976      C
977      C
978      C
979      C
980      C
981      C
982      C
983      C
984      C
985      C
986      C
987      C
988      C
989      C
990      C
991      C
992      C
993      C
994      C
995      C
996      C
997      C
998      C
999      C
1000      C

```

Table A-3 Subroutine CHLSKI

```

1  SUBROUTINE CHLSKI
2
3  C CHOLESKI DECOMPOSITION ( MATRIX INVERSION)
4
5  COMMON /KALIN2/ T1 T2 T3
6
7  COMMON /MATINV/ Z SZ
8
9  REAL T1(9,9)
10  F T2(3,3)
11  G T3(3,9)
12
13  C DOUBLE PRECISION Z(3,3)
14  A SZ(3)
15
16  DO 2000 I = 1,3
17  DO 1000 J = 1,3
18  Z(I,J) = T2(I,J)
19  CONTINUE
20  CALL DECOM
21  DO 5000 I = 1,9
22  DO 3000 J = 1,3
23  SZ(J) = T1(J,I)
24  CONTINUE
25  CALL SUBS
26  DO 4000 J = 1,3
27  T3(J,I) = SZ(J)
28  CONTINUE
29  DO 5000
30  CONTINUE
31  RETURN
32  END
33
1000
2000
3000
4000
5000
C

```

Table A-4 Subroutine CORECT

```

1  SUBROUTINE CORECT
2
3  C UPDATE EXTRAPOLATION WITH MEASUREMENTS
4  C
5  COMMON /KALID / RESUL,BARN ,SPS ,PKAL ,QA ,RMEAS,QKAL
6  A
7  COMMON /KALIN2/ T1 ,T2 ,T3
8  C
9  REAL RESUL(3)
10 .BARN(3)
11 A
12 .SPS(9)
13 B
14 .PKAL(9,9)
15 C
16 .QA(3,3)
17 D
18 .RMEAS(3,3)
19 E
20 .QKAL(9,9)
21 F
22 .ZMEAS(3)
23 G
24 .HKAL(3,9)
25 H
26 .PHI(9,9)
27 I
28 .RMEAS(3,3)
29 J
30
31 REAL T1(9,9)
32 T1(3,3)
33 T2(3,3)
34 T3(3,9)
35
36 COMPUTE HKAL * PKAL STORE IN T1
37
38 DO 3000 K = 1,3
39 DO 2000 I = 1,9
40 T1(K,I) = 0.0
41 DO 1000 J = 1,9
42 T1(K,I) = T1(K,I) + HKAL(K,J) * PKAL(J,I)
43 CONTINUE
44 CONTINUE
45 CONTINUE
46
47 1000
48 2000
49 3000
50
51 RMEAS + HKAL * PKAL * HKAL(TRANPOSE) STORE IN T2
52
53 DO 6000 K = 1,3
54 DO 5000 I = 1,3
55 T2(K,I) = RMEAS(K,I)
56 DO 4000 J = 1,9
57 T2(K,I) = T2(K,I) + T1(K,J) * HKAL(J,I)
58 CONTINUE
59 CONTINUE
60 CONTINUE
61
62 CHLSKI COMPUTES T2 INVERSE * T1 STORE IN T3
63
64 CALL CHLSKI
65
66 CORRECT PKAL ( NOTE PKAL SYMETRIC )
67
68 DO 9000 K = 1,9

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

57 DO 8000 I = 1,9
58 DO 7000 J = 1,3
59 PKAL(K,I) = PKAL(K,I) - T1(J,K) * T3(J,I)
60 CONTINUE
61 CONTINUE
62 CONTINUE
63 CONTINUE
64 STABILIZE PKAL
65
66 DO 9700 I = 1,9
67 PKAL(I,I) = AMAX1(PKAL(I,I),0.000001)
68 DO 9500 J = 1,9
69 PKAL(J,I) = 0.5*(PKAL(I,J) + PKAL(J,I))
70 CONTINUE
71 CONTINUE
72
73 INNOVATION ( ZMEAS - HKAL * SPS ) STORE IN RESDUL
74
75 DO 11000 I = 1,3
76 RESDUL(I) = ZMEAS(I)
77 DO 10000 J = 1,9
78 RESDUL(I) = RESDUL(I) - HKAL(I,J) * SPS(J)
79 CONTINUE
80 CONTINUE
81
82 CORRECT SPS ( T3 TRANSPOSE IS THE KALMAN GAIN )
83
84 DO 13000 I = 1,9
85 DO 12000 J = 1,3
86 SPS(I) = SPS(I) + T3(J,I) * RESDUL(J)
87 CONTINUE
88 CONTINUE
89
90 RETURN
91 END

```

Table A-5 Subroutine CTRF

```

1  SUBROUTINE CTRF
2
3  C COMPUTES COORDINATE TRANSITION MATRIX, TRV, FROM REFERENCE TO FRENET FRAME
4  C AND FRENET FRAME ACCELERATION ERROR COVARIANCE - QAP
5
6  COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,OKAL
7  .ZMEAS ,HKAL ,PHI ,RMEAS
8
9  COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
10
11
12 COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE,SIGMR
13 .STDAZ1,STDAZ2,STDEL1,STDEL2
14
15 REAL RESDUL(3)
16 .BARN(9)
17 .SPS(9)
18 .PKAL(9,9)
19 .QA(3,3)
20 .RMEASI(3,3)
21 .OKAL(9,9)
22 .ZMEAS(3)
23 .HKAL(3,9)
24 .PHI(9,9)
25 .RMEASI(3,3)
26
27 REAL QAP(3,3)
28 .VMAG
29 .TRV(3,3)
30 .ADOTVU
31 .AP
32 .GN
33 .RSPS
34 .TRS(3,3)
35 .THRUST
36
37 REAL PVAL(9)
38 .DELTAT
39 .CONE
40 .CTWO
41 .CTHREE
42 .SIGMR
43 .STDAZ1
44 .STDAZ2
45 .STDEL1
46 .STDEL2
47
48 VMAG = SORT ( SPS(4)*SPS(4) + SPS(5)*SPS(5) + SPS(6)*SPS(6) )
49 VMAGP = SORT ( SPS(4)*SPS(4) + SPS(5)*SPS(5) )
50
51 CHECK FOR NEAR ZERO VELOCITY
52 IF ( VMAGP .LT. 50.0 ) GO TO 3000
53
54 FAST TARGET
55
56

```

[illegible]

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

114      GN = 0.0
115      C
116      CONTINUE
117      GO TO 4000
118      CONTINUE
119      C
120      C
121      C
122      SLOW TARGET (VELOCITY SMALL)
123      DO 3500 I = 1.3
124      DO 3250 J = 1.3
125      TRV(I,J) = 0.0
126      QAP(I,J) = 0.0
127      CONTINUE
128      TRV(I,1) = 1.0
129      QAP(I,1) = (CONE*DELTAT)**2
130      CONTINUE
131      3500
132      4000
133      C
134      RETURN
135      END

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-6 Subroutine CTRS

```

1  SUBROUTINE CTRS
2
3  C COMPUTES TRANSITION MATRIX, TRS, FROM REFERENCE TO ESTIMATED L.O.S FRAME
4  C
5  COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,OKAL
6  COMMON /ZMEAS ,HKAL ,PHI ,RMEAS
7  C
8  COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
9  C
10 REAL RESDUL(3)
11 ,BARN(9)
12 ,SPS(9)
13 ,PKAL(9,9)
14 ,QA(3,3)
15 ,RMEASI(3,3)
16 ,OKAL(9,9)
17 ,ZMEAS(3)
18 ,HKAL(3,9)
19 ,PHI(9,9)
20 ,RMEAS(3,3)
21 C
22 REAL QAP(3,3)
23 ,VMAG
24 ,TRV(3,3)
25 ,ADOTVU
26 ,AP
27 ,GN
28 ,RSPS
29 ,TRS(3,3)
30 ,THRUST
31 C
32 RSPS = SQRT ( SPS(1)**2 + SPS(2)**2 + SPS(3)**2 )
33 RSPSP = SORT ( SPS(1)**2 + SPS(2)**2 )
34 C
35 CHECK FOR NEAR ZERO RANGE
36 IF ( RSPSP .LT. 5.0 ) GO TO 1000
37 C
38 RANGE OK
39 C
40 TRS(1,1) = SPS(1) / RSPS
41 TRS(1,2) = SPS(2) / RSPS
42 TRS(1,3) = SPS(3) / RSPS
43 TRS(2,1) = - SPS(2) / RSPSP
44 TRS(2,2) = SPS(1) / RSPSP
45 TRS(2,3) = 0.0
46 TRS(3,1) = - SPS(1)*SPS(3) / ( RSPS * RSPSP )
47 TRS(3,2) = - SPS(2)*SPS(3) / ( RSPS * RSPSP )
48 TRS(3,3) = RSPSP / RSPS
49 GO TO 4000
50 CONTINUE
51 1000
52 C
53 RANGE SMALL
54 C
55 DO 3000 I = 1,3
56

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

57 DO 2000 J = 1.3
58 TRS(I,J) = 0.0
59 CONTINUE
60 TRS(I,I) = 1.0
61 CONTINUE
62 CONTINUE
63 CONTINUE
64 RETURN
65 END

2000
3000
4000
C

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDG

Table A-7 Subroutine DECOM

```

1  SUBROUTINE DECOM
2
3  C DECOMPOSE (MATRIX INVERSION)
4  C
5  COMMON /MATINV/ Z ,SZ
6  C
7  DOUBLE PRECISION Z(3,3)
8  ,SZ(3)
9  A
10 C
11 DO 3000 I = 1,3
12   K = I - 1
13   DO 2000 J = 1,3
14     IF ( K.EQ. 0 ) GO TO 1500
15     DO 1000 M = 1,K
16       Z(I,J) = Z(I,J) - Z(M,J) * Z(M,I)
17     CONTINUE
18   IF ( (J-I).EQ. 0 ) GO TO 1750
19   Z(I,J) = Z(I,J) / Z(I,I)
20   GO TO 1875
21 CONTINUE
22 Z(I,J) = DSORT( DABS(Z(I,J)) )
23 CONTINUE
24 CONTINUE
25 CONTINUE
26 C
27 RETURN
28 END

```

Table A-8 Subroutine EXTRAP
SUBROUTINE EXTRAP

```

1  C
2  C EXTRAPOLATES STATE VECTOR AND STATE COVARIANCE MATRIX
3  C
4  C
5  C
6  C
7  C
8  C
9  C
10 C
11 C
12 C
13 C
14 C
15 C
16 C
17 C
18 C
19 C
20 C
21 C
22 C
23 C
24 C
25 C
26 C
27 C
28 C
29 C
30 C
31 C
32 C
33 C
34 C
35 C
36 C
37 C
38 C
39 C
40 C
41 C
42 C
43 C
44 C
45 C
46 C
47 C
48 C
49 C
50 C
51 C
52 C
53 C
54 C
55 C
56 C

COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,QKAL
,ZMEAS ,HKAL ,PHI ,RMEAS

REAL RESDUL(3)
BARN(9)
SPS(9)
PKAL(9,9)
QA(3,3)
RMEASI(3,3)
QKAL(9,9)
ZMEAS(3)
HKAL(3,9)
PHI(9,9)
RMEAS(3,3)

REAL ST(9)
TEMP(9,9)

PREDICT STATE VECTOR

DO 2000 I = 1,9
ST(I) = 0.0
DO 1000 J = 1,9
ST(I) = ST(I) + PHI(I,J) * SPS(J)
CONTINUE
DO 3000 I = 1,9
SPS(I) = ST(I)
CONTINUE
DO 3500 I = 1,9
SPS(I) = SPS(I) + BARN(I)
CONTINUE

PKAL * PHI (TRANPOSE) STORE IN TEMP

DO 6000 K = 1,9
DO 5000 I = 1,9
TEMP(K,I) = 0.0
DO 4000 J = 1,9
TEMP(K,I) = TEMP(K,I) + PKAL(K,J) * PHI(I,J)
CONTINUE
CONTINUE
CONTINUE

PREDICT PKAL AS QKAL + PHI * PKAL * PHI(TRANPOSE)

DO 9000 K = 1,9
DO 8000 I = 1,9
PKAL(K,I) = QKAL(K,I)
DO 7000 J = 1,9
PKAL(K,I) = PKAL(K,I) + PHI(K,J) * TEMP(J,I)
CONTINUE
CONTINUE
CONTINUE

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

57 8000 CONTINUE
58 9000 CONTINUE
59 C
60 C
61 RETURN
62 END

Table A-9 Subroutine FILTER

1	C	SUBROUTINE FILTER
2	C	KALMAN ESTIMATOR OF POSITION, VELOCITY, AND ACCELERATION
3	C	COMPUTE TRANSITION ERROR COVARIANCE MATRIX QKAL
4	C	AND TRANSITION ERROR MEAN BARN
5	C	CALL CTRF
6	C	CALL NSAR
7	C	CALL Q
8	C	EXTRAPOLATE STATE
9	C	CALL EXTRAP
10	C	COMPUTE MEASUREMENT ERROR COVARIANCE MATRIX
11	C	CALL CTRS
12	C	CALL R
13	C	UPDATE STATE AND STATE ERROR COVARIANCE
14	C	CALL CORECT
15	C	RETURN
16	C	END
17		
18		
19		
20		
21		
22		
23		
24		
25		
26		

THIS PAGE IS EXCLUDED
FROM COPY FURNISH

```

1 SUBROUTINE INTFLT
2
3 C INITIALIZATION OF KALMAN ALGORITHMS
4 C
5 COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,QKAL
6 ,ZMEAS ,HKAL ,PHI ,RMEAS
7 C
8 COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
9 ,TRS ,THRUST
10 C
11 COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE,SIGMR
12 ,STDAZ1,STDAZ2,STDEL1,STDEL2
13 C
14 REAL RESDUL(3)
15 ,BARN(9)
16 ,SPS(9)
17 ,PKAL(9,9)
18 ,QA(3,3)
19 ,RMEASI(3,3)
20 ,QKAL(9,9)
21 ,ZMEAS(3)
22 ,HKAL(3,9)
23 ,PHI(9,9)
24 ,RMEAS(3,3)
25 C
26 REAL QAP(3,3)
27 ,VMAG
28 ,TRV(3,3)
29 ,ADOTVU
30 ,AP
31 ,GN
32 ,RSPS
33 ,TRS(3,3)
34 ,THRUST
35 C
36 REAL PVAL(9)
37 ,DELTAT
38 ,CONE
39 ,CTWO
40 ,CTHREE
41 ,SIGMR
42 ,STDAZ1
43 ,STDAZ2
44 ,STDEL1
45 ,STDEL2
46 C
47 COMMON /NMNVAL/
48 PVAL(1) = 9000000.0
49 PVAL(2) = 9000000.0
50 PVAL(3) = 4000000.0
51 PVAL(4) = 10000.0
52 PVAL(5) = 10000.0
53 PVAL(6) = 10000.0
54 PVAL(7) = 400.0
55 PVAL(8) = 400.0
56

```


THIS PAGE IS BEST Q
FROM COPY FURNISHED

114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-11 Subroutine LAMDAS

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56

C SUBROUTINE LAMDAS
C UPDATE LAMDA AND CAPLAM
C
C COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,QKAL
C ,ZMEAS ,HKAL ,PHI ,RMEAS
C
C COMMON /ABIO / LAMDA ,CAPLAM
C
C DOUBLE PRECISION RESDUL(3)
C .BARN(9)
C .SPS(9)
C .PKAL(9,9)
C .QA(3,3)
C .RMEASI(3,3)
C .QKAL(9,9)
C .ZMEAS(3)
C .HKAL(3,9)
C .PHI(9,9)
C .RMEAS(3,3)
C
C DOUBLE PRECISION LAMDA(9)
C .CAPLAM(9,9)
C
C DOUBLE PRECISION SAB(9,9)
C .TEMAB1(9,9)
C .TEMAB2(9)
C .TEMAB3(9,9)
C .TAB3(9)
C .TAB4(9,9)
C .TAB5(9,9)
C
C UPDATE LAMDA
C
C SAB = HKAL(TRANPOSE) * RMEASI * HKAL
C
C DO 4000 I = 1,9
C DO 3000 J = 1,9
C SAB(I,J) = 0.0
C DO 2000 M = 1,3
C DO 1000 N = 1,3
C SAB(I,J) = SAB(I,J) + HKAL(M,I) * RMEASI(M,N) *
C HKAL(N,J)
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C
C COMPUTE I - PKAL * SAB STORE IN TEMAB1
C
C DO 7000 I = 1,9
C DO 6000 J = 1,9
C TEMAB1(I,J) = 0.0
C TEMAB3(I,J) = 0.0
C DO 5000 K = 1,9

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

57 TEMAB1(I,J) = TEMAB1(I,J) - PKAL(I,K) * SAB(K,J)
58 CONTINUE
59 CONTINUE
60 CONTINUE
61 DO 7500 I = 1.9
62 TEMAB1(I,I) = TEMAB1(I,I) + 1.0
63 TEMAB3(I,I) = TEMAB3(I,I) + 1.0
64 CONTINUE
65 7500
66 C
67 C
68 PHI(TRANPOSE) * LAMDA STORE IN TEMAB2
69 DO 9000 I = 1.9
70 TEMAB2(I) = 0.0
71 DO 8000 J = 1.9
72 TEMAB2(I) = TEMAB2(I) + PHI(J,I) * LAMDA(J)
73 CONTINUE
74 8000
75 9000
76 C
77 C
78 TEMAB2 - HKAL(TRANPOSE) * RMEASI * RESDUL STORE IN TAB3
79 DO 14000 I = 1.9
80 TAB3(I) = TEMAB2(I)
81 DO 13000 J = 1.3
82 DO 12000 K = 1.3
83 TAB3(I) = TAB3(I) - HKAL(J,I) * RMEASI(J,K) * RESDUL(K)
84 CONTINUE
85 CONTINUE
86 12000
87 13000
88 14000
89 C
90 C
91 C
92 LAMDA = TEMAB1(TRANPOSE) * TAB3
93 DO 16000 I = 1.9
94 LAMDA(I) = 0.0
95 DO 15000 J = 1.9
96 LAMDA(I) = LAMDA(I) + TEMAB1(J,I) * TAB3(J)
97 CONTINUE
98 15000
99 16000
100 C
101 C
102 C
103 UPDATE CAPLAM
104 PHI * TEMAB1 STORE IN TAB4
105 DO 18000 I = 1.9
106 DO 17000 J = 1.9
107 TAB4(I,J) = 0.0
108 DO 16500 K = 1.9
109 TAB4(I,J) = TAB4(I,J) + PHI(I,K) * TEMAB1(K,J)
110 CONTINUE
111 CONTINUE
112 16500
113 17000
114 18000
115 C
116 C
117 C
118 TAB4(TRANPOSE) * CAPLAM * TAB4 STORE IN TAB5
119 DO 22000 I = 1.9
120 DO 20000 J = 1.9
121 TAB5(I,J) = 0.0
122 DO 20000 M = 1.9

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

114 DO 19000 N = 1.9
115 TAB5(I,J) = TAB5(I,J) + TAB4(M,I) * CAPLAM(M,N) *
116 TAB4(N,J)
117 CONTINUE
118 CONTINUE
119 CONTINUE
120 CONTINUE
121 C
122 C
123 C
124
125 DO 25000 I = 1.9
126 DO 24000 J = 1.9
127 CAPLAM(I,J) = TAB5(I,J)
128 DO 23000 K = 1.9
129 CAPLAM(I,J) = CAPLAM(I,J) + SAB(I,K) * TEMAB1(K,J)
130 CONTINUE
131 CONTINUE
132 CONTINUE
133 RETURN
134 END

```

Table A-12 Subroutine NBAR

```

1  SUBROUTINE NBAR
2  C COMPUTES STATE TRANSITION ERROR VECTOR BARN
3  C
4  COMMON /KALTO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,OKAL
5  A ,ZMEAS ,HKAL ,PHI ,RMEAS
6  C
7  COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
8  A ,TRS ,THRUST
9  C
10 COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE,SIGMR
11 A ,STDAZ1,STDAZ2,STDEL1,STDEL2
12 C
13 REAL RESDUL(3)
14 A ,BARN(3)
15 B ,SPS(9)
16 C ,PKAL(9,9)
17 D ,QA(3,3)
18 E ,RMEASI(3,3)
19 F ,QKAL(9,9)
20 G ,ZMEAS(3)
21 H ,HKAL(3,9)
22 I ,PHI(9,9)
23 J ,RMEAS(3,3)
24 C
25 REAL QAP(3,3)
26 A ,VMAG
27 B ,TRV(3,3)
28 C ,ADOTVU
29 D ,AP
30 E ,GN
31 F ,RSPS
32 G ,TRS(3,3)
33 H ,THRUST
34 C
35 REAL PVAL(9)
36 A ,DELTAT
37 B ,CONE
38 C ,CTWO
39 D ,CTHREE
40 E ,SIGMR
41 F ,STDAZ1
42 G ,STDAZ2
43 H ,STDEL1
44 I ,STDEL2
45 C
46 IF ( VMAG .LT. 50.0 ) GO TO 1000
47 TBARN1 = -GN*2 / VMAG *DELTAT
48 TBARN2 = ADOTVU *GN /VMAG *DELTAT
49 TBARN3 = 0.0
50 GO TO 2000
51 1000 CONTINUE
52 TBARN1 = 0.0
53 TBARN2 = 0.0
54 TBARN3 = 0.0
55 2000 CONTINUE
56

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

TRANSFORM TO REFERENCE FRAME

BARN(7) = TRV(1,1)*TBARN1 + TRV(2,1)*TBARN2 + TRV(3,1)*TBARN3
BARN(8) = TRV(1,2)*TBARN1 + TRV(2,2)*TBARN2 + TRV(3,2)*TBARN3
BARN(9) = TRV(1,3)*TBARN1 + TRV(2,3)*TBARN2 + TRV(3,3)*TBARN3

BARN(4) = BARN(7) * 0.5 * DELTAT
BARN(5) = BARN(8) * 0.5 * DELTAT
BARN(6) = BARN(9) * 0.5 * DELTAT

BARN(1) = BARN(7) * 0.166667 * DELTAT**2
BARN(2) = BARN(8) * 0.166667 * DELTAT**2
BARN(3) = BARN(9) * 0.166667 * DELTAT**2

RETURN
END

57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73

C
C
C

C

C

C

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-13 Subroutine Q

```

1  SUBROUTINE Q
2
3  C COMPUTES STATE TRANSITION ERROR COVARIANCE MATRIX OKAL
4  C
5  COMMON /KALIO / RESDUL,BARN,SPS,PKAL,QA,RMEASI,OKAL
6  COMMON /ZMEAS,HKAL,PHI,RMEAS
7
8  C COMMON /KALUT / QAP,VMAG,TRV,ADOTVU,AP,GN,RSPS
9  COMMON /TRS,THRUST
10
11 C COMMON /KALDAT/ PVAL,DELTAT,CONE,CTWO,CTHREE,SIGMR
12 COMMON /STDAZ1,STDAZ2,STDEL1,STDEL2
13
14 REAL RESDUL(3)
15 BARN(9)
16 SPS(9)
17 PKAL(9,9)
18 QA(3,3)
19 RMEASI(3,3)
20 OKAL(9,9)
21 ZMEAS(3)
22 HKAL(3,9)
23 PHI(9,9)
24 RMEAS(3,3)
25
26 C REAL QAP(3,3)
27 VMAG
28 TRV(3,3)
29 ADOTVU
30 AP
31 GN
32 RSPS
33 TRS(3,3)
34 THRUST
35
36 C REAL TEMP(3,3)
37
38 C REAL PVAL(9)
39 DELTAT
40 CONE
41 CTWO
42 CTHREE
43 SIGMR
44 STDAZ1
45 STDAZ2
46 STDEL1
47 STDEL2
48
49 C TRANSFORM ACCELERATION COVARIANCE MATRIX (QAP) TO
50 REFERENCE FRAME (QA)
51
52 C QAP * TRV STORE IN TEMP
53
54 DO 2000 I = 1,3
55 DO 1000 J = 1,3
56 TEMP(I,J) = 0.0

```

```

57 QA(I,J) = 0.0
58 CONTINUE
59 CONTINUE
60 DO 5000 I = 1,3
61 DO 4000 K = 1,3
62 DO 3000 J = 1,3
63 TEMP(I,K) = TEMP(I,K) + QAP(I,J) * TRV(J,K)
64 CONTINUE
65 CONTINUE
66 CONTINUE
67 CONTINUE
68 TRV(TRANSPOSE) * TEMP STORE IN QA
69
70 DO 8000 I = 1,3
71 DO 7000 K = 1,3
72 DO 6000 J = 1,3
73 QA(I,K) = QA(I,K) + TRV(J,I) * TEMP(J,K)
74 CONTINUE
75 CONTINUE
76 CONTINUE
77 CONTINUE
78
79 TRANSITION ERROR COVARIANCE MATRIX QKAL
80
81 DO 10000 I = 1,3
82 DO 9000 J = 1,3
83 QKAL(I,J) = QA(I,J)*DELTA**4/36.
84 QKAL(I,J+3) = QA(I,J)*DELTA**3/12.
85 QKAL(I,J+6) = QA(I,J)*DELTA**2 /6.0
86 QKAL(I+3,J) = QA(I,J)*DELTA**3/12.
87 QKAL(I+3,J+3) = QA(I,J)*DELTA**2 /4.0
88 QKAL(I+3,J+6) = QA(I,J)*DELTA /2.0
89 QKAL(I+6,J) = QA(I,J)*DELTA**2 /6.0
90 QKAL(I+6,J+3) = QA(I,J)*DELTA /2.0
91 QKAL(I+6,J+6) = QA(I,J)
92 CONTINUE
93 CONTINUE
94 CONTINUE
95 RETURN
END

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDO

Table A-14 Subroutine R

1	C	1	COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEASI,QKAL
2	C	2	COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE,SIGMR
3	C	3	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
4	C	4	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
5	C	5	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
6	C	6	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
7	C	7	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
8	C	8	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
9	C	9	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
10	C	10	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
11	C	11	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
12	C	12	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
13	C	13	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
14	C	14	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
15	C	15	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
16	C	16	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
17	C	17	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
18	C	18	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
19	C	19	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
20	C	20	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
21	C	21	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
22	C	22	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
23	C	23	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
24	C	24	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
25	C	25	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
26	C	26	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
27	C	27	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
28	C	28	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
29	C	29	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
30	C	30	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
31	C	31	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
32	C	32	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
33	C	33	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
34	C	34	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
35	C	35	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
36	C	36	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
37	C	37	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
38	C	38	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
39	C	39	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
40	C	40	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
41	C	41	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
42	C	42	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
43	C	43	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
44	C	44	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
45	C	45	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
46	C	46	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
47	C	47	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
48	C	48	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
49	C	49	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
50	C	50	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
51	C	51	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
52	C	52	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
53	C	53	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
54	C	54	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
55	C	55	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
56	C	56	COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS

COMPUTE COVARIANCE MATRIX IN ESTIMATED L.O.S FRAME

SIGRXS = SIGMR**2

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```

57 SIGRYS = (SDAZ1)**2+ ( RSPS+STDAZ2)**2
58 SIGRZS = (STDEL1)**2 + (RSPS+STDEL2)**2
59 DO 2000 I = 1,3
60 DO 1000 J = 1,3
61 RLOS(I,J) = 0.0
62 RLOSI(I,J) = 0.0
63 RMEAS(I,J) = 0.0
64 RMEASI(I,J) = 0.0
65 TEMP1(I,J) = 0.0
66 TEMP2(I,J) = 0.0
67 CONTINUE
68 1000
69 CONTINUE
70 RLOS(1,1) = SIGRYS
71 RLOS(2,2) = SIGRYS
72 RLOS(3,3) = SIGRZS
73 RLOSI(1,1) = 1.0 / SIGRYS
74 RLOSI(2,2) = 1.0 / SIGRYS
75 RLOSI(3,3) = 1.0 / SIGRZS
76
77 TRANSFORM COVARIANCE MATRIX, RLOS, TO REFERENCE FRAME
78
79 RLOS * TRS, STORE IN TEMP1
80
81 DO 5000 I = 1,3
82 DO 4000 K = 1,3
83 DO 3000 J = 1,3
84 TEMP1(I,K) = TEMP1(I,K) + RLOS(I,J) * TRS(J,K)
85 TEMP2(I,K) = TEMP2(I,K) + RLOSI(I,J) * TRS(J,K)
86 CONTINUE
87 CONTINUE
88 CONTINUE
89
90 TRS(TRANPOSE) * TEMP1
91
92 DO 8000 I = 1,3
93 DO 7000 K = 1,3
94 DO 6000 J = 1,3
95 RMEAS(I,K) = RMEAS(I,K) + TRS(J,I) * TEMP1(J,K)
96 RMEASI(I,K) = RMEASI(I,K) + TRS(J,I) * TEMP2(J,K)
97 CONTINUE
98 CONTINUE
99 CONTINUE
100 6000
101 7000
102 8000
103 CONTINUE
104 RETURN
105 END

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-15 Subroutine SUBS

```

SUBROUTINE SUBS
C
C RECURSIVE INVERSION
C
COMMON /MATINV/ Z ,SZ
C
DOUBLE PRECISION Z(3,3)
,SZ(3)
A
C SZ(1) = SZ(1) / Z(1,1)
C
C FORWARD SUBSTITUTION
C
DO 2000 I = 2,3
K = I - 1
DO 1000 J = 1,K
SZ(J) = SZ(I) - Z(J,I) * SZ(J)
CONTINUE
SZ(I) = SZ(I) / Z(I,I)
CONTINUE
SZ(3) = SZ(3) / Z(3,3)
DO 4000 I = 2,3
K = I - 1
J1 = 4 - I
BACK SUBSTITUTION
C
C DO 3000 J = 1,K
C JT = 4 - J
C SZ(J1) = SZ(J1) - SZ(JT) * Z(J1,JT)
C CONTINUE
3000 SZ(J*) = SZ(J1) / Z(J1,J1)
4000 CONTINUE
C
RETURN
END

```

DISTRIBUTION LIST
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCPA-S 5001 Eisenhower Avenue Alexandria, VA 22333	2	Chief Defense Logistics Studies Information Exchange US Army Logistics Management Center ATTN: DRXMC-D FT Lee, VA 23801
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-R 5001 Eisenhower Avenue Alexandria, VA 22333	1	HQDA (SAUS-OR) WASH DC 20310
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-AD 5001 Eisenhower Avenue Alexandria, VA 22333	2	HQDA (DAMO-RQA) COL Campbell/LTC Solomon WASH DC 20310
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-OR 5001 Eisenhower Avenue Alexandria, VA 22333	2	Commander US Army Concepts Analysis Agency ATTN: MOCA-MR MOCA-ADL 8120 Woodmont Avenue Bethesda, MD 20014
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-SE 5001 Eisenhower Avenue Alexandria, VA 22333	1	Commander US Army Operational Test and Evaluation Agency ATTN: CSTE-STS-L 5600 Columbia Pike Falls Church, VA 22041
2	Commander Defense Documentation Center ATTN: TCA Cameron Station Alexandria, VA 22314	4	Commander US Army Armament Research and Development Command ATTN: DRDAR-SE DRDAR-LCS DRDAR-SCF DRDAR-SCS DRCPM-ADG Dover, NJ 07801
		2	Commander US Army Aviation Research and Development Command ATTN: DRDAV-BCA DRDAV-BCE PO Box 209 St. Louis, MO 63166

DISTRIBUTION LIST (Continued)
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Director Research and Technology Laboratories ATTN: DAVDL Moffet Field, CA	1	Commander US Army Troop Support and Aviation Readiness Command ATTN: DRSTS-BA 4300 Goodfellow Boulevard St. Louis, MO 63120
1	Commander US Army Electronics Research and Development Command ATTN: DRDEL-AP-OA (B. Zarwyn) 2800 Powder Mill Road Adelphi, MD 20783	2	Commander US Army Combat Developments Experimentation Command ATTN: ATEC-PL-M ATEC-PL-TL FT Ord, CA 93941
8	Commander US Army Missile Research and Development Command ATTN: DRDMI-X DRDMI-DS DRDMI-CD DRDMI-CW DRDMI-RSI DRDMI-Y DRCPM-HEL DRCPM-ROC Redstone Arsenal, AL 35809	2	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-PL ATAA-TG White Sands Missile Range, NM 88002
1	Commander US Army Tank Automotive Research and Development Command ATTN: DRDTA-V Warren, MI 48090	1	Commandant US Army Air Defense School ATTN: ATSA-CD-A FT Bliss, TX 79916
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L (Tech Lib) Rock Island, IL 62199	1	Commander US Army Aviation Center and Fort Rucker ATTN: ATZQ-CD FT Rucker, AL 36362
3	Commander US Army Missile Materiel Readiness Command ATTN: DRSMI-D DRCPM-CF DRCPM-HA Redstone Arsenal, AL 35809	1	President US Army Air Defense Board ATTN: ATZC FT Bliss, TX 79916
		2	Commander US Army White Sands Missile Range ATTN: STEWS-TE-ML STEWS-NR-PA White Sands Missile Range, NM 88002

DISTRIBUTION LIST (Continued)
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Commander US Army Combined Arms Combat Developments Activity ATTN: ATCA-CA Fort Leavenworth, KS 66027	1	Director Institute for Defense Analysis ATTN: Dr. J. Ross 400 Army Navy Drive Arlington, VA 22202
1	Commandant US Military Academy ATTN: COL John L. Palmer Dept. of Engineering West Point, NY 10996	1	United States Air Force Academy Department of Astronautics and Computer Science ATTN: DFACS/LTC E. J. Bauman USAF Academy, CO 80840
1	Commander Office of Naval Research ATTN: Mr. S.M. Selig 800 North Quincy Street Arlington, VA 22217		<u>Aberdeen Proving Ground</u> Dir, AMSAA ATTN: D.O'Neill, AWD H. Peaker, AWD J. Meredith, AWD R. Scheder, AWD H. Burke, CSD T. Perkins, CSD S. Johnson, MSD
1	Director Special Projects Office Department of the Navy Washington, DC 20360		Cdr, TECOM ATTN: DRSTE-CT-A DRSTE-CM-A DRSTE-AD-A DRSTE-AD-S DRSTE-AD-M
1	Commander Naval Weapons Center ATTN: Code 753 China Lake, CA 93557		Dir, HEL ATTN: DRXHE-SP
1	AF/XOFA Washington, DC 20330		Dir, BRL ATTN: DRDAR-BLB (H. Reed) DRDAR-BLB (P. Yip)
1	AF/SAV Washington, DC 20330		Dir, MTD ATTN: STEAP-MT-M
1	AFSC (DOVS) Andrews AFB Washington, DC 20331		
1	TAWC/OA (Mr. J. Durrenberger) Eglin AFB, FL 32542		
1	AFWL (SA) Kirtland AFB, NM 87117		